République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Mohamed Khider Biskra

N° d'ordre :.....

Série :.....

Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie

Département d'Informatique

THÈSE

Présentée pour obtenir le grade de

DOCTORAT EN SCIENCES EN INFORMATIQUE

Par

Hammadi BENNOUI

THEME

Distributed Causal Model-based Diagnosis: An Approach by Interacting Petri Nets

---

Soutenue le : 19 Janvier 2012

Devant le jury composé de :

M. Djamel Eddine Saidouni	Professeur à l'Université de Constantine	Président
M. Allaoua Chaoui	Professeur à l'Université de Constantine	Rapporteur
M. Kazar Okba	Professeur à l'Université de Biskra	Examinateur
M. Cherif Foudil	Maître de conférences A à l'Université de Biskra	Examinateur
M. Mustapha Bourahla	Maître de conférences A à l'Université de M'sila	Examinateur
M. Kamel Barkaoui	Professeur au CNAM de Paris, France	Invité

Acknowledgment

I would like to thank all the people that supported me in many ways, making possible this work.

First and foremost I want to thank Allaoua Chaoui and Kamel Barkaoui, my supervisors, for their support and their encouragement. I felt so lucky to work with them, not only for their deep knowledge and expertise, but because they are so beautiful people. This thesis is the result of the work done together.

Special thanks to Mourad Maouche and Mohamed Bettaz who introduced me in the field of model-based diagnosis and Petri nets.

Finally, I would like to thank Djamel Eddine Saidouni, Kazar Okba, Cherif Foudil and Mustapha Bourahla for reviewing and evaluating this thesis.

Contents

List of Figures

Chapter 1

Introduction

The problem of fault diagnosis of artifact systems has received a great attention during the last two decades due to its importance in terms of safety and efficiency of operation. Numerous complementary approaches have been proposed, based on the level of detail chosen for the model of the system to be diagnosed and the kinds of faults that need to be diagnosed. The starting point of these approaches is to model the structure and/or behavior of the system to be diagnosed. When an abnormal behavior of the system is observed, the diagnosis consists to tracking back on the model for explaining such a misbehavior. Accordingly, a traditional diagnostic system can be viewed as a *centralized* system having a model of the whole system to be diagnosed and receiving all observation signalizations. There are, however, several reasons why in some applications such a single agent approach may be inappropriate. First of all, if the system is physically distributed and large, e.g. modern telecommunication networks, there may be not enough time to compute a diagnosis centrally and to communicate all observations. Secondly, if the structure of the system is dynamic, e.g. AGV systems driving in a platoon, it may change too fast to maintain an accurate global model of the system over time. Finally, sometimes a central model is simply undesirable. For example, if the system is distributed over different legal entities, one entity does not wish other entities to have a detailed model of its part of the system. For such systems, a *distributed* approach of multiple diagnostic agents might offer a solution.

The model (knowledge) of a system can be distributed over the agents in two principally

different ways[1] (cf. [79]):

- *spatially distributed*: knowledge of system behavior is distributed over the agents according to the spatial distribution of the system's components, and
- *semantically distributed*: knowledge of system behavior is distributed over the agents according to the type of knowledge, e.g. a separate model of the electrical and of the thermodynamical behavior of the system.

For both types of distributions, a multiagent system can establish the same global diagnoses as a single diagnostic agent having the combined knowledge of all agents [76].

We focalize ourselves in this thesis to the problem of spatially distributed causal model-based diagnosis. We consider the system to be diagnosed as a collection of interacting subsystems in which when a fault occurs in one subsystem, it may generate some fault indications (i.e. symptoms) and may propagate to the neighborhood. The diagnostic system itself is defined as a set of diagnostic agents each of which is associated with a specific subsystem. In particular, each agent has a local model of the assigned subsystem and may receive observations generated only by elements of this subsystem. The local model describes the causal behavior of the subsystem as well as its interactions within adjacent ones. When agents observe an aberrant behavior, each one is charged to explain the received local observation on the basis of its local model. As a result, each diagnostic agent calculates a set of local diagnoses. In causal models, the diagnoses are to be given in terms of initial states that explain the set of observed symptoms using the cause-effect relationships described in the model. Such initial states represent the initial perturbations leading the system to behave abnormally.

Since each agent has a limited knowledge about the whole system to be diagnosed, it may be possible that local diagnoses of different agents are inconsistent when they are considered altogether. In order to ensure the required consistency and to guarantee that such local diagnoses recover completely global ones that would be obtained by a centralized agent having a global view of the whole system, agents should communicate among them to reject inconsistent diagnoses.

[1] Combinations are, of course, also possible.

We use in this thesis a particular class of Petri nets called "*Behavioral Petri Nets*" (*BPNs*) which has been proposed in [2] to represent the causal behavior of a system for centralized diagnosis purposes. In particular, the causal behavior of each subsystem is described by a local BPN model and interactions among subsystems are captured through tokens that may pass via common bordered places between BPNs. Diagnostic reasoning scheme may be accomplished by exploiting classical analysis techniques of Petri nets. More particularly, diagnosis can be implemented locally by a backward analysis (BW-Analysis) of the corresponding reachability graph to explain the received local observation. The BW-Analysis exploits two different types of tokens (*normal* and *inhibitor* tokens) aimed at modeling the truth or falsity of the condition associated with a marked place. This allows as to point out inconsistencies when looking for possible explanation of a given marking in a BPN. As a result, each agent obtains a set of local initial markings from which diagnoses have to be given. Then, to achieve the consistency with the local diagnoses of all other agents, each one requests from its neighbors the required marking of its bordered places for each computed diagnosis. At this step, agents receiving such a request will construct their reachability graphs in a forward fashion to check if the requested marking of bordered places is reachable from at least one of their computed initial markings. If so, the local diagnosis from which the exchanged message has been generated is considered globally consistent; otherwise, it is not supported by diagnoses of the neighborhood and consequently it must be discarded.

Accordingly, such reasoning suffers from the so-called state space explosion problem even for small net models. This is due to the utilization of reachability graphs as a basis on which analysis is accomplished especially in the consistency checking phase where several graphs may be constructed by each agent. In order to face such a problem, we may exploit algebraic analysis techniques, known also as invariant analysis, which are shown useful in [4, 67, 68] for improving complexity in centralized diagnostic reasoning based on Petri nets versus reachability graph analysis. In particular, we concentrate in our work on the distributed analysis of P-invariants of the net models which are generated in an off-line manner. More specifically, we require that each diagnostic agent utilizes the set of minimal supports of its P-invariants to implement local preliminary computation as well as to check the required consistency of its local diagnoses with those of the other agents. Thus, the set of minimal

supports of P-invariants may be considered as a pre-compiled structure of the system model on which diagnosis is implemented. The idea of using a compiled structure of the system model to the on-line diagnosis is borrowed from Sampath *et al.* [77] in their work on discrete event systems (DES) diagnosis. They propose to generate from a finite state automaton describing the system model another automaton termed the *Diagnoser* which encompasses more information about the system state (i.e. information about the presence or absence of faults). The Diagnoser is used to both test the diagnosability properties of the system and perform on-line monitoring of the system for the purpose of diagnosis which necessitates a synchronization between the Diagnoser and the system model. Thus, the invariant based diagnosis is similar to the Diagnoser approach regarding the off-line pre-compilation of the system model to face the complexity problem during on-line diagnosis. However, several features make our proposals different from that of [77] and others, namely we use causal models in which the observations to be explained are modeled as partial states of the system to be diagnosed and not as observable events of the Diagnoser approach. Similarly, the faults in terms of which diagnoses have to be given are considered as initial states which have no causes in the causal model and not as unobservable transitions adopted in the context of DES diagnosis. Another difference is that we do not require that such a compiled structure will be synchronized with the system model which is one of the key features of the approach of [77] and all its extensions [38, 43, 51]. It is to be noted however that in this thesis we do not treat the question of diagnosability analysis and we concentrate only on how to implement diagnosis by distributed analysis of interacting BPNs.

This dissertation is structured in six chapters divided into two parts: the state of the art and contributions. The state of the art part presents background on model-based diagnosis according to logical points of view as well as Petri nets ones. The contribution part contains our proposals for diagnosing distributed systems by analyzing interacting BPNs.

We begin in the following chapter by formulating the diagnostic problem. The chapter concentrates on the model-based view of diagnosis. It is devoted to synthesize the different works that are made in the field. The attention is focused firstly on logical formalizations and on declarative characterizations; and secondly on procedural aspects that characterize different diagnostic engines.

Chapter 3 surveys diagnosis on Petri nets models. It starts with a brief outline of basic notions related to Petri nets as a model for concurrent systems, then it discusses the problem of DES diagnosis as well as DES diagnostic methods that have prevailed in the literature. The chapter concludes with a classification of DES diagnosis implementations according to topological point of view.

The first chapter of the contributions part (chapter 4) attempts to formalize the notion of distributed model-based diagnosis. It defines the diagnostic system as a multi-agent system that reflects the same network structure of the system to be diagnosed. In particular, the chapter characterizes the diagnosis of each agent in the system as well as the distribution of the process of diagnosis over the different agents of the diagnostic system.

In chapter 5, we show how model-based diagnosis of distributed system can be accomplished by BPN analysis based on reachability graphs. We start by formalizing the system model as a set of place bordered BPNs each of which model the causal behavior of one subsystem, then we show how the BW analysis of each BPN implements local diagnosis of the associated subsystem based on its causal model. Once local diagnoses are obtained, global consistency between them will be ensured through a cooperation protocol among the diagnostic agents. It is to be noted that a preliminary version of such a distributed technique has been published as a conference paper [6].

Based on the drawbacks of the distributed BW analysis (object of chapter 5), chapter 6 proposes the definition of the invariant based technique as an alternative to implement causal model-based diagnosis of distributed systems. In particular, after characterizing diagnostic solutions by the minimal supports of the nets P-invariants the chapter discusses algorithmically local calculations made by each agent as well as how to exploit such supports during consistency checking phase. The invariant based diagnosis technique has been published in [7].

Chapter 7 considers the problem of relationships that may exist between symptoms. This requires in one hand the expression of such relationships in the system model; and on the other hand the adaptation of the analysis techniques to handle these relations. More particularly, the chapter proposes a novel set of backward firing rules to account for precedence relationships among symptom signalizations. Besides such an adaptation, the chapter con-

siders also the case where some of the signaled symptoms have been lost or suppressed. If so, it may be possible that the given diagnostic problem is inconsistent; and the retained solution consists to restore the required consistency to the given problem by slightly changing the given observation so that the problem admits an interpretation model. Some parts of chapter 7 contents have been occurred in [5].

Finally, we conclude the thesis by summarizing what we consider as the main contributions of our work as well as their limitations. Perspective works are identified on the basis of such limitations.

Part I

State of the art

Chapter 2

Model-based diagnosis

2.1 Introduction

Advances in modern design and manufacturing technology have enabled us to build systems of high complexity. When these systems fail to function correctly, they need to be repaired. The repair process pass necessarily through a diagnosis step for locating those subsystems that are responsible for the observed malfunction.

Since the complexity of diagnosis increases with increasing design complexity, the efficient automation of this task becomes essential and gives rise to an important area of computer science. Especially in the area of artificial intelligence, big efforts have been spent in the attempt to define approaches leading to the automatic diagnosis of broken systems. Such efforts have resulted in the proposition of two fundamentally different families of approaches to diagnostic reasoning [48]. One is based on "*heuristic-based*"expert systems, the other is based on "*model-based*"ones.

In the first, heuristic-based approaches, one attempts to codify diagnostic rules of thumb and past experience of human diagnosticians considered experts in the involved domain. Representatives of these approaches are *Expert Systems of the First Generation*, with the blood infection diagnosis system MYCIN [80] as the famous instance. Here, diagnostic skills are captured by sets of more or less direct associations between observable symptoms and diseases as their potential causes. Being grounded in experience gained in previous cases, diagnosis was treated as collecting empirical evidence for the presence of certain malfunctions

rather than a strict deduction process. The necessity to state diagnostic knowledge in terms of explicit symptom-fault associations inherently limited the scope of applicability. Only the identification of previously encountered faults was possible based on previously observed symptoms of systems that are well experienced to allow for the enumeration of the relevant associations. Because these associations tend to be quite specific for narrow types of systems, building such systems was a matter of time-consuming single-piece production [31, 48].

All this turned out to be too restrictive when confronted with requirements in diagnosis of technical systems. Industrial application of automated diagnosis has to cover the detection and localization of new kinds of faults, exhibited by newly designed and constructed systems and the interpretation of symptoms never observed before. The diagnosis of such systems stems from knowledge about the physical and technological principles underlying the (intended or deviating) functioning of these systems which allows one to systematically deduce fault hypotheses from available observations even if the system is novel.

The key idea of the second family of approaches to diagnosis, often called diagnosis from the first principles or model-based diagnosis, is to explicitly represent this knowledge as a model of the system structure and/or behavior of its constituents and to organize diagnosis as an inference process based on this model and the observed behavior. This view created the demand for and the possibility of developing a rigorous theoretical foundation for automated diagnosis. In particular, this comprises a formal characterization of the goal and of the inferences that achieve the goal, given model-based predictions and the actual observations of the system's behavior. Early introduced, model-based diagnostic systems, referred to as *Second Generation Expert Systems* such as those proposed in [27, 28, 74], provide declarative system-independent representation languages and system-independent diagnostic procedures. As a consequence of this independence, they are capable, using hierarchical representations [58], of diagnosing complex systems in any domain. Moreover, they are more robust than heuristic-based systems, because they can deal with unexpected cases not covered by heuristic rules. In addition, their knowledge bases are less expensive to create and flexible in regard to design changes since they are a straightforward representation of designs. They do not require rule verification, which can be serious problem in writing heuristic rules. However, the drawback of model-based diagnostic systems is that they require

more complex computation, and hence they are generally not efficient as heuristic-based ones which can guide efficient diagnosis for known cases.

Different attempts to combine heuristic-based and model-based approaches have been made by some researchers [20, 33, 48]. The searched goal in these works consists to get benefit of the advantages of both approaches, namely the robustness of model-based reasoning in one hand and the simplicity of computation and efficiency characterizing heuristic-based one at the other hand.

The overall goal of this chapter is to survey the foundations of model-based diagnosis. In a little more than twenty years, work on automated diagnosis based on models has managed both to establish a strong theoretical basis and to create a technology mature enough to tackle real industrial applications. This does not only allow us to build application systems with formally stated preconditions and provable capabilities and properties. It also provides challenges for theoretical work and hard criteria for evaluating its results and helps to focus it. In fact, model-based diagnosis becomes really one of the rare success stories in artificial intelligence [31].

We begin, in the following section by presenting a formulation of the diagnostic problem within a model-based reasoning view. It introduces some notations that will be used to present, in section 2.3, the main diagnostic approaches that have prevailed in the literature, namely: consistency-based and abductive approaches to diagnosis. A unified approach which integrates these two approaches is outlined. Details concerning the content of the model of the system to be diagnosed are presented in section 2.4. Section 2.5 and 2.6 give the logical characterization of the diagnostic problem and of the set of solutions to such a problem. Procedural aspects of the inference mechanisms proposed in different diagnostic systems are synthesized in section 2.7. Finally, section 2.8 summarizes the main concepts presented in this chapter.

2.2 Problem formulation

Starting with the logical framework given in [74], we make use, in this section, of the first-order logic with equality as a language of knowledge representation.

2.2.1 System model

Unlike the heuristic-based approaches to diagnosis, model-based approaches are proposed to diagnose broken systems independently of the application domain. That is why, a general domain-independent concept of a system description is indispensable. The following definition of a system description introduced by Reiter in [74] has been considered by most model-based diagnostic frameworks. It is designed to formalize as abstractly as possible the concept of a component, and the concept of a collection of interacting components.

Definition 1 *A system description SD for a system S is a pair (BM, COMPS), where:*

1. *BM, the behavioral model, is a set of first-order formulas representing the knowledge about S;*

2. *COMPS, the system components, is a finite set of constants.*

In all intended applications, the behavioral model will mention a distinguished unary predicate AB(.), interpreted to mean "*abnormal*". The argument of such a predicate necessarily belongs to $COMPS$. However, as mentioned in [74] it is possible to introduce several kinds of AB(.) predicates and represent more general component properties.

The use of AB predicate for describing a system is borrowed from McCarty [55] who exploits such a predicate in conjunction with his formalization of circumscription to account for various patterns of nonmonotonic common-sense reasoning.

2.2.2 Observation

Real-world diagnostic settings involve observations (or measurements). Without observations, we have no way of determining whether something is wrong and hence whether a diagnosis is called for.

Definition 2 (from [74]) *An observation OBS for a system S is a finite set of first-order formulas defining I, the inputs, and O, the outputs of S.*

Notice that distinguished inputs and outputs are features of many man-made artifacts, such as electronic devices. In other applications, however, the observation OBS may not

define a set of inputs. It specifies only a set of findings as observable elements of the system to be diagnosed.

It should be also noted that OBS does not specify all outputs of the system S, nor that SD is a complete description (the behavior of each component is not assumed to be completely specified). The only assumption in most diagnostic frameworks is that the given knowledge is consistent. In other words, $(BM \cup OBS)$ admits a particular model, the so-called *intended interpretation* [32], which represents the actual problem universe.

2.2.3 Diagnoses

Suppose that we have determined that a system $S = (BM, \{c_1, ..., c_n\})$ is faulty, by which we mean informally that we have made an observation OBS which conflicts with the way the system is meant to behave if all its components behave correctly.

The repair of S will pass necessarily through a diagnosis step for explaining the observed malfunction. It consists to find a subset of components – say, $\Delta \subseteq COMPS$ – which, when assumed to be failed, will explain why the system exhibits a misbehavior. Thus, a diagnosis can be defined intuitively as a conjunction that certain of the components are faulty and the rest normal. Now the main problem is to specify which components we conjuncture to be faulty in order to provide an explanation for the observed malfunction.

To reach such specification, a model-based diagnosis is guided by the interaction between observations and predictions conducted separately from the system model (Figure 2-1). Thus, it relies solely on the system model BM, its components $COMPS$, and the observation OBS. In particular, it does not use any heuristic information about the system failures gained by the experience, for example, of the kind "when the system exhibits such and such aberrant behavior, then in 90% of these cases, such and such components have failed".

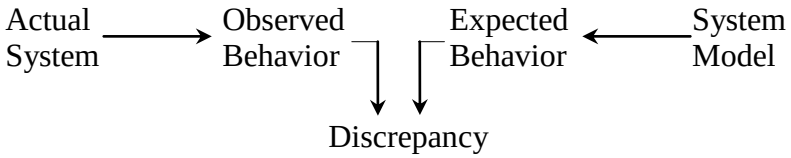

Figure 2-1: Diagnosis as the interaction of observation and expectation.

2.2.4 Notations

When computing diagnoses, we are interested in discovering formulas where AB is the only predicate symbol which occurs. Therefore, let L_{AB} be a first-order language which contains all the formulas which we can built with the AB predicate symbol alone.

Notation 1 *A formula (respec. clause, respec. conjunction, respec. literal) from the language L_{AB} is referred to as an ab-formula (respec. ab-clause, respec. ab-conjunction, respec. ab-literal).*

Any diagnosis will be represented as an ab-conjunction which does not contain two occurrences of the same component. In particular, a diagnosis is a satisfiable ab-conjunction. Note that there exists only a finite number of ab-formulas (up to logical equivalence). Moreover, we adopt the following notation:

Notation 2

- *The set of components occurring in an ab-conjunction Δ is denoted by $\mathbf{C}(\boldsymbol{\Delta})$;*
- *The set of positive literals in Δ is denoted by $\boldsymbol{\Delta}^{+}$;*
- *The set of negative literals in Δ is denoted by $\boldsymbol{\Delta}^{-}$.*

The dual concept of prime implicant and prime implicate in first-order logic play a central role in most formalizations of model-based diagnosis. Let us present their definitions [29, 32].

Definition 3 *Let I_c be an existentially quantified conjunction of literals, then I_c is an implicant of a closed formula F if $I_c \vdash F$. Let PI_c be an implicant of a closed formula F, then PI_c is a prime implicant of F if $PI_c \vdash F$ and, for any purely existential conjunction of literals PI'_c, if $PI'_c \vdash F$ and $PI_c \vdash PI'_c$ then $PI'_c \vdash PI_c$.*

Definition 4 *Let I_d be a universally quantified disjunction of literals, then I_d is an implicate of a closed formula F if $F \vdash I_d$. Let PI_d be an implicate of a closed formula F, then PI_d is a prime implicate of F if $F \vdash PI_d$ and, for any purely universal disjunction of literals PI'_d, if $F \vdash PI'_d$ and $PI'_d \vdash PI_d$ then $PI_d \vdash PI'_d$.*

2.3 Approaches to model-based diagnosis

The ultimate objective of the diagnostic reasoning is to determine the state of each component of the system to be diagnosed. Formally, we have from [27, 28, 29, 74]:

Definition 5 *The actual diagnosis, Δ, is an ab-conjunction such that:*

- Δ *is complete: the state* $AB(c)$ *or* $\neg AB(c)$ *of each component* c *is given;*
- Δ *holds in the intended interpretation.*

Unfortunately, due to the incompleteness of our knowledge about the system, it is not always possible to compute Δ in a purely deductive way. By deduction only a set of *partial* diagnoses, $\Delta_{j_1}, ..., \Delta_{j_n}$, is usually generated. The term partial means that the state of all components is not determined.

Since the actual diagnosis is complete and holds in the intended interpretation[1], then Δ implies any ab-formula that holds in that interpretation. In particular, among the sets of partial diagnoses, there is at least one partial diagnosis Δ_{j_i} that can be extended to account for the state of all components. Formally, we have: $\Delta \vdash \Delta_{j_i}$.

Thus, computing diagnoses turns out to be:

1. selecting some partial diagnoses, and
2. extending them to set the state of a maximum number of components with a maximal confidence.

Once again, this contributes to support the thesis that *diagnostic reasoning is by nature hypothetico-deductive* [32]. This point of view has leads researchers to define some preference criteria for characterizing the hypotheses that should be assumed when extending partial diagnoses.

Different formalizations of this notion of diagnostic reasoning have been proposed in the literature. The first attempts to such formalizations have generally considered two extremes of the diagnosis problem:

[1] In fact, Δ can be regarded as the ab-part of the intended interpretation.

1. There is knowledge about how components are structured and work normally. There is no knowledge as to how malfunctions occur and manifest themselves. Diagnosis consists of isolating deviations from normal behavior. This has normally been the preserve of an approach termed *consistency-based diagnosis* [23, 27, 74].

2. We have just information on faults (diseases) and their symptoms, and want to account for abnormal observations. This has traditionally been the preserve of a second approach called *abduction-based diagnosis* [14, 16, 21, 66].

In the following, we attempt to provide the principles of each of these approaches with an emphasize on their preference criteria, then we present a unified definition that will be used throughout this thesis.

2.3.1 Consistency-based diagnosis

Since it is proposed originally to deal with models of the correct behavior, consistency-based diagnosis [23, 27, 30, 74] is oriented towards diagnosing systems with the following requirement: *there always exist characteristic manifestations to be observed when the system works normally.* The components of systems such as electronic devices meet this feature since the expected outputs of each component can be expressed as a function of its inputs. In this approach, the behavioral model is constructed according to the following methodology [31]:

1. for each component c of $COMPS$ that could be faulty, we have the hypothesis $\neg\text{AB}(c)$;

2. we write as facts implications that state what follows from assumptions of normality.

Suppose that $S = (BM, COMPS)$ is a system under diagnosis and let $O(I)$ be the expected outputs of S. This can be formalized by:

$$BM \cup I \cup \{\neg\text{AB}(c) \mid c \in COMPS\} \vdash O(I)$$

Suppose that there is a discrepancy between O, the observed outputs, and $O(I)$. Then, one can conclude that the system S is behaving incorrectly. Indeed, assume that:

$$BM \cup I \cup \{\neg_{\mathrm{AB}}(c) \mid c \in COMPS\} \vdash (O(I) \Rightarrow \neg O)$$

Then, necessarily we have:

$$BM \cup OBS \cup \{\neg_{\mathrm{AB}}(c) \mid c \in COMPS\} \text{ is inconsistent.}$$

As it has noted in [18, 27, 32, 74], the objective of consistency-based diagnosis is to explain this inconsistency which stems from the assumption $\{\neg_{\mathrm{AB}}(c) \mid c \in COMPS\}$, i.e. that all components are behaving correctly.

Consequently, Δ such that $C(\Delta^-) = COMPS$ is not the actual diagnosis since any presumed diagnosis must be consistent with the given knowledge. A possible diagnosis Δ for S must be an ab-conjunction such that $BM \cup OBS \cup \{\Delta\}$ is consistent [27, 29, 31, 32, 74].

Note that such diagnosis contributes to explain why the observed outputs differ from the expected ones when assuming that all components are normal.

Preference criteria

In this paragraph, we present the two preference criteria used in consistency-based diagnosis for refining the set of diagnoses.

A. Maximizing the number of described components

Among the possible diagnoses, the *preferred* ones are those which set the state of the maximal number of components. Hence, they are exactly the possible ones.

As we have mentioned, in most cases the incompleteness of the knowledge prevents us from obtaining the complete possible diagnoses in a purely deductive way. Hence, obtaining a complete possible diagnosis Δ generally requires to make additional hypotheses. Formally, if Δ is a complete possible diagnosis, then according to [74] we have:

$$BM \cup OBS \cup \{\neg_{\mathrm{AB}}(c_i) \mid c_i \in C(\Delta^-)\} \vdash \wedge_{c_j \in C(\Delta^+)} \mathrm{AB}(c_j)$$

Generally, there are several complete possible diagnoses. Hence, more selection is needed for refining those diagnoses. This calls for the following second criterion.

B. Minimizing the set of abnormal components

Among the complete possible diagnoses, the preferred ones are those including the minimal sets of abnormal components. More precisely, if Δ_1 and Δ_2 are two complete possible diagnoses and $C(\Delta_1^+)$ is a subset of $C(\Delta_2^+)$ then Δ_1 will be preferred to Δ_2.

Thus, if Δ is a complete possible diagnosis, then Δ is minimal if among all complete diagnoses, $C(\Delta^+)$ is minimal w.r.t set inclusion.

In fact, the minimality criterion is nothing but it is the formal expression of the *parsimony principle* proposed in [74]. Probability theory argues in favor of this criterion when for each component the probability of failure is lower than the probability of correct behavior.

Characterizing and computing diagnoses

Our objective in this paragraph is to show how to determine all diagnoses for a malfunctioning system and to present a logical characterization of complete possible and minimal diagnoses. There is a direct generate-and-test mechanism based upon the consistency required in this approach: systematically generate subsets Δ of $COMPS$, generating Δs with minimal cardinality first, and test the consistency of

$$BM \cup OBS \cup \{\neg_{\text{AB}}(c) \mid c \in COMPS - \Delta\}$$

As it has been noted in most papers of model-based diagnosis, the previous problem with this mechanism is that it is too inefficient for systems with large number of components. Instead, Reiter in [74] proposes a method based upon a suitable formalization of the concept of a conflict, a concept due originally to de Kleer [27].

In [29, 31], a conflict for a diagnostic problem is defined as an ab-clause entailed by $BM \cup OBS$. A positive conflict is a conflict all of whose literals are positive. In other words, a positive conflict is a subset $C \subset COMPS$ which cannot all be functioning correctly; i.e. $BM \cup OBS \cup \{\neg_{\text{AB}}(c) \mid c \in C\}$ is inconsistent. More precisely, a conflict is any ab-clause which is an implicate of $BM \cup OBS$.

To achieve the set of diagnoses for a broken system, three fundamental subtasks will be explored [24]:

- *generating* hypotheses by reasoning from a symptom to a positive conflict (i.e. to a collection of components whose misbehavior may plausibly have caused that symptom);
- *testing* each hypothesis to see whether it can account for all observations of system behavior; then
- *discriminating* among those that survive testing.

Thus, the concept of conflicts provide an intermediate step in determining the diagnoses and are central to most diagnostic frameworks.

We should now present the following characterization which has been introduced in [29, 32]. It defines the set of complete possible and minimal diagnoses in terms of prime implicants.

Characterization 1 *Let $DD+$ be the set of all positive conflicts of S, the system to be diagnosed. The diagnosis Δ such that $C(\Delta^+) = \{c_{j_1}, c_{j_2}, ..., c_{j_m}\}$ is a complete possible and minimal diagnosis iff $AB(c_{j_1}) \wedge AB(c_{j_2}) \wedge \cdots \wedge AB(c_{j_m})$ is a prime implicant of $DD+$.*

In [29], it has been shown that the minimal complete and possible diagnoses cannot be considered as a basis for describing the complete possible diagnoses. It has also been proved that changing the status of a component from normal to abnormal in a complete possible diagnosis does not necessarily result in a possible diagnosis.

Adequacy of preference criteria

In the consistency-based approach any computed diagnosis does not exactly explain the observed behavior of the system. In fact the observations, uniformly embedded in the rest of the knowledge, only contribute to prove the inconsistency of the system when assuming that all components are normal. In other words, the observed actual behavior is not significant. What is important here is that this observed behavior differs from the expected one. That is why the consistency-based approach leads sometimes to undesirable results as it is illustrated in the domain of digital circuits by [32].

2.3.2 Abductive diagnosis

Abductive diagnosis [14, 16, 21, 66] is proposed originally to deal with fault behavioral models (i.e. models of the faulty behaviors of the system to be diagnosed). It views the world in terms of causes and effects. The methodology followed in this approach is [66]:

1. The possible hypotheses are the possible causes (faults, diseases) parametrized by the values which they depend;

2. We axiomatize how symptoms follow from causes. These axioms should be facts if the symptom is always present given the cause and be possible hypotheses otherwise.

Because the correct behavior of the system to be diagnosed is not modeled, its expected outputs cannot be predicted. Hence, it is not possible to detect any discrepancy between the observed outputs and the expected ones. Unlike, the consistency-based approach, in the abductive approach $BM \cup OBS \cup \{\neg \text{AB}(c)\}$ remains consistent.

Since there is no inconsistency to explain when the expected normal manifestations are unavailable, diagnostic reasoning is confined to giving some account for some observed manifestations.

Let CO be a combination of outputs to be explained. The abductive diagnosis for CO is defined to be an ab-conjunction Δ such that:

- $BM \cup I \cup \Delta \vdash CO$ and

- $BM \cup I \cup \Delta$ is consistent; where $CO \subseteq O$.

Let us present the two preference criteria which are used in this approach to refine the set of diagnoses.

Preference criteria

A. Maximizing the explained outputs

In the abduction-based approach to diagnosis, the selection of preferred diagnoses appeals to the *confirmation principle* [17]: the preferred diagnoses are those which explain a maximal set

of manifestations. However, not all manifestations are equally significant and an interesting problem is the selection of a pertinent subset of manifestations to be explained.

Clearly, the confidence we have in some abductive diagnosis increases with the number of explained outputs.

B. Minimizing the abnormality

Among the abductive diagnoses for a given combination of outputs CO, the preferred ones are those including a minimum number of abnormal components. In other words, let Δ be an abductive diagnosis for CO, then Δ is minimal if among all abductive diagnoses for CO, $C(\Delta)$ is minimal w.r.t set inclusion.

This time, the probability theory gives evidence for this criterion. Indeed, consider Δ and Δ' two abductive diagnoses for CO such that Δ implies Δ'. By this criterion, we prefer the minimal one Δ' which is more probable than Δ.

Characterizing and computing diagnoses

In [29, 31], a characterization of the abductive view for diagnosis is provided. It relates also the abductive diagnoses to the notion of prime implicants.

Characterization 2 *The abductive diagnoses for CO are exactly the positive implicants of $(BM \wedge I) \Longrightarrow CO$ which are consistent with $BM \cup I$. The minimal abductive diagnoses for CO are exactly the positive ab-prime implicants of $(BM \wedge I) \Longrightarrow CO$ which are consistent with $BM \cup I$.*

Clearly, any abductive diagnosis is an extension of at least one minimal abductive diagnosis. Nevertheless, not every extension of a minimal abductive diagnosis is an abductive one since this operation can lead to an inconsistency with $BM \cup I$.

Adequacy of preference criteria

The main drawback of the abduction-based diagnostic approach, as it is defined above, is the following: the actual diagnosis is not necessarily an extension of a minimal abductive diagnosis. More precisely, any abductive diagnosis is not necessarily a possible one. As in

[32], considering the definition of an abductive diagnosis Δ, one can conclude that $BM \cup I \cup \Delta \cup CO$ is consistent.

Nevertheless, any abductive diagnosis is necessarily a possible when $CO = O$. In particular, every complete abductive diagnosis for O is a complete possible diagnosis [14, 17]. Unfortunately, as it is shown by [32], an abductive diagnosis for O does not always exist.

As one can easily gather from these presentations, consistency-based and abductive diagnosis differ in the representation about normality and faults and in the meaning they give to "*explain*". Typically, according to the consistency-based view, a component is abnormal if its observed behavior deviates from the expected one; while in the abductive view, a component is abnormal if it manifests as it is described in the behavioral model (in fact, BM will describe according to the abductive approach different faulty behavioral modes. In the discussion above we have assumed, for reasons of simplicity, that only one faulty behavioral mode, noted AB, is modeled). The difference of explanation becomes obvious, for the consistency-based diagnosis, a solution explains why the system exhibits a malfunction; while in the abductive one, a solution attempts to explain why the system reacts as it is observed.

2.3.3 The integration approach

In these views of model-based diagnosis, the link between consistency-based reasoning and models of the correct behavior and the one between abductive reasoning and faulty models seem to be a natural choice: if one has a theory (the fault model) that can predict the observations, then the notion of covering is the "right"notion of explanation; while if the observations are only the "negation"of the predictions of the theory (the model of the correct behavior), then consistency is the "right"notion of explanation.

Over the last two decades, some attempts to break such privileged links have been made [17, 28, 32] and the advantages of combining fault models and models of the correct behavior have been recognized by some researchers [17, 18, 66]. The approaches to such a combination can be classified into two main groups [18]:

- extensions of consistency-based diagnosis to deal with fault models [28, 30, 31];

- extensions of abductive diagnosis to deal with models of the correct behavior [16, 66].

In almost all these approaches the integration is very homogeneous: the correct and faulty behaviors of a system have been represented in a uniform way and very few changes to the original reasoning schemes have been made.

Consequently, Console and Torasso in [18] analyze the two former approaches of diagnosis at their logical definitions. They tried to propose a unified framework based on the integration of consistency-based and abductive reasoning rather than extending one of them. In particular, they single out the existence of a spectrum of alternatives in the logical definition of diagnosis by reformulating each of the two notions of diagnostic problem as an abduction one with consistency constraints. The alternatives in such a spectrum range from purely consistency-based approaches (such as the one proposed by de Kleer and Williams [27]) to purely abductive approaches (such as the one proposed by Poole [66]).

Since this spectrum appears as a general framework, having the two approaches presented above as particular cases, we attempt in the following to describe with more details the principle concepts that characterize this framework. In our proposals presented in the second part, we make adaptation of these concepts for diagnosing multiple failures in distributed systems.

2.4 What's in *BM*?

According to the unified approach outlined above, the set of normal and faulty behaviors of the system to be diagnosed should be described in a uniform way using a suitable language. In this approach, each component of $COMPS$ is characterized by a set of behavioral modes. Such characterization was introduced by Holtzblatt [41] in his generalization of the GDE system (General Diagnostic Engine) [27] and used by de Kleer and Williams in their SHERLOCK's system [28].

In SHERLOCK, the behavior of the system to be diagnosed S can be represented as the consequences of the behavioral modes of its constituents. In particular, each component c_i is associated with a set of behavioral modes $\{correct, fault_{i_1}, ..., fault_{i_n}\}$ (where *correct* corresponds to the correct behavior of the component and each one of the values $fault_{i_j}$

corresponds to a distinguished faulty behavior of the component). Notice that one of such faulty modes could be the "unknown mode" with which no model is associated (see [28]). The union of the sets of behavioral modes of all components of S are denoted by the set of *abducible symbols*. The reason for such a name will be clear in the following. As we shall see, in fact the fundamental problem of diagnosis is that of determining, given a set of observations, the behavioral modes of the components of S "explaining" the observations. This means that the abducible symbols are the basic elements of such "explanations". Given an abducible symbol α, the fact that a component c is in mode α is represented by the atom $\alpha(c)$.

Given the behavioral modes of the components $COMPS$ of S and the consequences of such modes, one can build the system description SD which specifies the structure and behavior of S as discussed in [28]. The structure of S specifies the components and their interconnections. Components are described as being in one of the set of its distinct modes, where each mode captures a physical manifestation of the component. The behavior of each component is characterized by describing its behavior in each of its distinct modes.

In [18], the behavioral model BM is formed by a set of Horn clauses in which the set of predicate symbols are partitioned into the two subclasses of the *abducible* and *non-abducible* symbols. The abducible symbols do not appear in the head of any clause in BM. This corresponds to assuming that the behavioral modes of the components are "primitive" concepts (in the sense that they cannot be defined in terms of other concepts).

As a simple example, consider the problem of modeling the behavior of a digital circuit containing and-gates [40]. Let us assume that we distinguish three different behavioral modes of an and-gate, i.e. that its set of behavioral modes is $\{correct, stuck\_at\_0, stuck\_at\_1\}$; the behavioral model of the circuit will contains the formulae:

$and\_gate(X) \wedge correct(X) \wedge inp_1(X, X_1) \wedge inp_2(X, X_2) \longrightarrow out(X, f_{AND}(X_1, X_2)$

$and\_gate(X) \wedge stuck\_at\_0(X) \longrightarrow out(X, 0)$

$and\_gate(X) \wedge stuck\_at\_1(X) \longrightarrow out(X, 1)$

where $f_{AND}(X_1, X_2)$ is the logical AND of X_1 and X_2. Notice that $\{correct, stuck\_at\_0, stuck\_at\_1\}$ is the set of abducible symbols in such a model and those conditions which appear only in the body of a clause in BM and which are not abducibles (such as, for

example, inp_1 and inp_2) may correspond to contextual conditions; we shall return more precisely to this point in the next section.

In the discussion above, BM is assumed to describe a model of the structure and behavior of the system to be diagnosed. However, all the discussion can be applied also to the case where the structure is not modeled at all and BM is a "*causal model*"of the behavior of the system under diagnosis. Causal models have been widely adopted in model-based diagnosis since the work of Weiss *et al.* in CASENET [89] and Patil in ABEL [61].

In causal models, the behavior of a system is characterized by a set of states (in fact, *partial states*, that is, entities that partially describe a situation in which the modeled system can be at a given time); each of which is in turn characterized by a finite set of *admissible values*, and relations among these states (i.e. cause-effect transformations among instances of states). For diagnostic purposes, [16] indicates that it is useful to distinguish among at least three types of states in the model:

- *Initial states*: they correspond to states which have no causes in the model. In the case of an abnormal behavioral model, they represent the initial perturbations leading the system to a given malfunction; and thus, they define the set of abducible symbols;
- *Internal states*: corresponding to the consequences of initial states. They are attached to system components that are not susceptible to make a part of a diagnosis, because they can be explained by the elements of initial states;
- *Manifestations*: corresponding to observable or measurable states and thus representing all expected symptoms in the case of faulty models.

In this view, performing a diagnosis means to explain a set of manifestations in terms of initial states, using the cause-effect relationships described in the model.

2.5 Characterizing diagnostic problems

In [18], the authors analyze in detail the notion of diagnostic problem. They argue that it is characterized by different types of data which must be treated in very different way in the diagnostic process.

In particular, a major distinction that they introduce is the one between contextual data and observations (such distinction has been originally proposed in [73] where the term "setting factor"is used to denote contextual data).

Contextual data are a set of parameters providing (contextual) information about the specific case under examination; typical examples are data such as the sex or the age of a patient (in medical diagnosis) or the "inputs"to a device (in other applications). Such data are very important since they allow the diagnostician to make predictions about the behavior of the system to be diagnosed (for example, the fact that a patient is a male allows a physician to exclude certain pathologies and to focus on other pathologies [18]). Typically, contextual data are known when a case is analyzed (or they can be easily gathered) and in some cases they are necessary to characterize the case itself. The important point is that contextual data need not to be accounted for by a diagnosis, but they are rather used to predict the behavior of the system to be diagnosed.

Data corresponding to observations, on the other hand, play a very different role (typical examples of observations are clinical findings or laboratory tests in a medical diagnosis, the outputs of a device in other applications). Observations are data that must be accounted for by a diagnosis. Now, let us present their definition of a diagnostic problem.

Definition 6 *A diagnostic problem DP is a triple $DP = \langle\langle BM, COMPS\rangle, Ctx, OBS\rangle$, where:*

- *$\langle BM, COMPS\rangle$ is the system description of the system to be diagnosed;*
- *Ctx is a set of ground atoms denoting the set of contextual data;*
- *OBS is a set of ground atoms denoting the set of observations to be explained.*

The meaning of each atom $f(a)$ in Ctx or in OBS is the following: in the specific problem to be solved the value a has been observed for the parameter f.

In this definition, different requirements are imposed by the authors. First, they impose the constraint that $Ctx \cup OBS$ can contain at most one instance of each symbol (i.e. an observable parameter cannot have more than one value). This corresponds to abstracting from time; i.e. to assuming that diagnosis is performed in a static environment (which is a

common assumption in most formalizations of model-based diagnosis, except some attempts to introduce the notion of time in diagnosis such as the works of [15, 89]). Second, they assume that all pieces of contextual information are known a priori (i.e. they are part of the data). Last, they impose, as it can be remarked from the definition, that contextual data and observations are represented by two distinguished sets of atoms.

2.6 Solving diagnostic problems

As we noticed previously, diagnosis can be characterized as the process of generating "explanations" for a set of observations in a given context. However, the term "explanation" has been used in the first model-based diagnostic systems with at least two different meanings (i.e. two different logical notions of "explanation").

- *explanation as consistency* (weak notion of explanation); in such a case a diagnosis explains an observation m if it does not contradict m (i.e. if it does not support $\neg m$);
- *explanation as covering* (strong notion of explanation); in such a case a diagnosis explains an observation m if it directly support m.

In most of the formalizations of diagnosis proposed in the literature, one of the two alternatives has been chosen. However, because the goal of the above characterization is of generalizing the other definitions of model-based diagnostic problem, an abstract definition which embeds the possibility of choosing among the two notions of explanation is indispensable. The following definition proposed in [18] is based on the reformulation of a diagnostic problem as an abduction problem with consistency constraints. Such reformulation contains a critical step: choosing which observations must be covered by a diagnosis. It is such a controversial choice that distinguishes among different definitions of diagnosis (of "explanation") and allows us to single out a spectrum of definitions.

Definition 7 *Given a diagnostic problem* $DP = \langle\langle BM, COMPS\rangle, Ctx, OBS\rangle$*, an abduction problem* AP *corresponding to* DP *is a triple* $AP = \langle\langle BM, COMPS\rangle, Ctx, \langle \Psi^+, \Psi^-\rangle\rangle$*, where:*

- $\Psi^+ \subseteq OBS$
- $\Psi^- = \{\neg f(x) \mid f(y) \in OBS$, *for each admissible value* x *of* f *other than* $y\}$

Ψ^+ is the set of observation atoms that must be covered by a solution; in principle, any subset of OBS can be chosen. Ψ^- on the other hand, is a set of negative literals and characterizes the set of values which conflict with the observation.

As we shall see, Ψ^- is used for consistency checking (Ψ^- is a set of denials and is interpreted as a set of consistency constraints that the solutions to the abduction problem must satisfy). Notice that Ψ^- may be an infinite set (in case at least one of the observable parameters can assume an infinite set of values).

The previous definitions take the assumption that the observation OBS characterizing a diagnostic problem is a set of atoms. This corresponds to assuming that definite knowledge about the observation is available. In some cases, however, it may be interesting to have the possibility of providing incomplete (partial) descriptions of the data characterizing a diagnostic problem. One way to specify incomplete knowledge about data was exposed in [16]. It consists to provide "negative"information about the observable parameters. Let us consider a negative information $\neg f(a)$: this can be regarded as a way to express that no definite knowledge about the actual value of the parameter f is available, but certainly f does not assume the value a (while any other value might a plausible).

In the following, we shall concentrate on the case where all the observation atoms are positive (definite); however, all the discussion can be easily generalized also to the case where "negative observations"are allowed. The definition of an abduction problem associated with a diagnostic problem can in fact be easily extended with positive (definite) and negative observations. In such a case, the atoms in Ψ^+ are a subset of the positive observations; while Ψ^- is the set of negative literals obtained as the union of the negative observations and the values that conflict with the positive ones.

We can now move to characterize the solutions to an abduction problem. The central task of abductive (diagnostic) reasoning is to identify those behavioral modes of the components whose consequences cover Ψ^+ (i.e. which predict Ψ^+) consistently with Ψ^-. More specifically, the space of hypotheses that has to be analyzed in order to determine the explanations for an

abduction problem AP is the space of the assignments of behavioral modes to the components $COMPS$ of the system. In particular, we have the following definition introduced in [28].

Definition 8 *Given a system description $\langle BM, COMPS\rangle$ and given the set of abducible symbols in BM, an assignment W for $COMPS$ is a set of ground abducible atoms such that for each $c \in COMPS$, W contains exactly one element of the form $\alpha(c)$ (where α is an abducible symbol).*

Notice that this corresponds to assuming that the behavioral modes of each component are mutually exclusive (which seems to be a reasonable assumption).

In particular, we are interested in those assignments that cover Ψ^+ consistently with Ψ^-.

Definition 9 *Given an abduction problem $AP = \langle\langle BM, COMPS\rangle, Ctx, \langle\Psi^+, \Psi^-\rangle\rangle$, an assignment W for $COMPS$ is an explanation for AP iff*

1. *W covers Ψ^+, that is for each $m \in \Psi^+$ we have that $BM \cup Ctx \cup W \vdash m$*
2. *W is consistent with Ψ^-, that is $BM \cup Ctx \cup W \cup \Psi^-$ is consistent. In other words, for each $\neg m \in \Psi^-$ we have that $BM \cup Ctx \cup W \nvdash m$*

Some remarks are worthwhile on such a definition. First of all, notice that consistency with Ψ^- corresponds to not predicting any value for an observable parameter different from the actual one (i.e. conflicting with the observed one). The notion of consistency used in this definition, therefore, is the same used in consistency-based definitions of diagnosis [23, 24, 27, 74]. Thus, this definition combines the two notions of explanation discussed previously: in order to provide a solution to an abduction problem (and thus to a diagnostic problem), an assignment must be consistent with all the observable parameters and must cover a selected group of parameters.

A second remark concerns the role played by contextual data. Such data are used to predict the expected behavior of the system and thus they play a role in consistency checking; however, they do not have to be covered. One way to enforce between contextual data and observations was proposed by Poole [66], who started from a simple example: given a system with input i and output o, how should one represent such data? Poole argued that there are

two alternative logical representations (namely $i \wedge o$ and $i \longrightarrow o$) and that different formalizations of diagnosis require different representations (more specifically, abductive diagnosis requires the representation of observation as an implication $i \longrightarrow o$; while consistency-based diagnosis requires the representation as a conjunction $i \wedge o$). Actually, such different representations can be regarded as a technical way to enforce the different roles played by the different types of data discussed above and captured, at the knowledge level, by the previous definition.

In general, given an abduction problem AP, there is more than one explanation for AP. Since one of the goals of diagnosis is to determine an explanation which minimizes abnormality, we can compare explanations by considering the sets of components which are assumed to be faulty in each explanation. More precisely, we can use the partition of each explanation into two subsets [29]:

- $correct(W) = \{correct(c) \mid correct(c) \in W\}$ where $correct$ is the linguistic term denoting the correct mode of the component c;
- $faulty(W) = W - correct(W)$

and then compare two explanations W_1 and W_2 by comparing the sets $faulty(W_1)$ and $faulty(W_2)$. We say that an explanation W is a minimal explanation if and only if the set $faulty(W)$ is minimal w.r.t set inclusion among the sets $faulty(W_i)$.

The same partition can be used to determine the solutions to a diagnostic problem: given a diagnostic problem DP, its associated abduction problem AP and an explanation W for AP, the set $faulty(W)$ is a solution to the diagnostic problem DP. This means that a solution to DP specifies which are the faulty components of the modeled system and which are the fault modes of these components that "explain"the observation.

2.7 Computational aspects

Up to here, the attention is focused essentially on logical formalizations and on declarative characterizations of the diagnostic problem and of the set of solutions to such a problem. The computational aspects of model-based diagnosis have not been treated in the above

description. In order to address more directly procedural aspects of physical system diagnosis, this section is devoted to synthesize the different inference mechanisms presented in the literature.

Early model-based diagnostic systems [23, 27, 28] are characterized by complicated inference strategies used to generate the set of solutions for a given problem. They make use of mechanisms similar to the ATMS one proposed in [27]. The ATMS[2] principle consists to propagate a set of assumptions on the given model of the system being diagnosed. In particular, since the diagnosis task is to identify the set of minimal conflicts that will be used in hypothesis generation step, the propagation of assumptions will identify all minimal conflicts of the given problem. In fact, a conflict can be identified by selecting some assumptions, referred to as an environment, and testing, according to the implemented notion of explanation, if they are inconsistent with the observation or they do not entail the observation. If they are, then the environment is a conflict. This requires an inference strategy $C(OBS, ENV)$ which, given the observation OBS made on the physical system and the environment ENV, determines whether the combination is consistent (respec. presents an entailment) [27]. All the first implemented systems use this principle with some adaptations concerning efficiency and simplicity.

In order to beat this complexity problem, more attention has been paid in the nineteen decade of last century to procedural aspects of physical system diagnosis. In particular, [19, 59, 65] propose a novel approach to the problem in which the diagnostic process is defined within a framework based on a Petri net model of the causal behavior of the system to be diagnosed. Indeed, the possibility of modeling causal relationships for describing the evolution of a system has been recognized as fundamental in order to guide the diagnostic system to explain a given set of symptoms [57].

The basic goal of the mentioned approach was to redefine the logical notion of a diagnostic problem in terms of reachability in the Petri net model. The key idea of these works consists of translating a set of definite clauses forming a logic program into a Petri net model, and using existing Petri net analysis methods to handle the diagnostic inference algebraically. More specifically, Portinale in [67] proposes an approach to the problem of performing diag-

[2]ATMS for Assumption based Truth Maintenance System.

nostic reasoning on a Petri net model by exploiting the notion of T-invariants. Its work is inspired from an idea presented in [59, 65] where T-invariant analysis is applied to the answer extraction problem in logic programming. Furthermore, in other papers [68, 69], P-invariant analysis and reachability graph analysis known in the Petri net theory have been applied in the same goal. In this way, a problem traditionally tackled using symbolic manipulation techniques can be partially reformulated in algebraic terms.

A performance evaluation between different implementations of the algebraic solutions in one hand and one based on classical inference mechanism has been exposed in [68]. It uses the running time consumed by each implementation as a comparison criterion. The evaluation shows that invariant approaches require short running time compared to the classical one to generate the same set of hypotheses. Furthermore, approaches based on reachability graph analysis of the Petri net model necessitate more considerable time to solve the diagnostic problem; but they are less complex than classical approaches, in addition to be suitable for parallel implementations [2, 69]. The evaluation concludes with the remark that Petri nets present challenging in the improvement of diagnostic reasoning process.

2.8 Conclusion

We began this chapter by formulating the diagnostic problem within a model-based reasoning view. The prevailed approaches to solve such a problem have been discussed. The discussion is focused on a comparative study of these approaches according to their preference criteria as well as their logical definitions.

An approach to the integration of consistency-based and abductive reasoning is studied in detail. Such study shows that a diagnostic problem is characterized by different types of data and that these types of data must be treated in a very different way in order to achieve the set of diagnoses. The chapter concludes with a discussion on the inference mechanisms used in different systems. Systems that make use of Petri nets formalism are shown to be less complex than those based on symbolic manipulations.

However, despite the considerable progress in developing a sound theoretical basis for automated diagnosis, there are a number of open issues that require more efforts. Some

generalizations seem possible to cover similar tasks, but there also exist some limitations that appear hard to overcome. In particular, systems with a behavior changing over time pose a number of hard problems. Besides the basis problem of modeling, which necessitates the handling of intermittent faults, we face a new dimension of complexity. Since the proposed approaches can diagnose only behavioral faults, the diagnosis of structural defects that establish new interaction paths between components is considered as an open problem.

Chapter 3

Diagnosis within Petri nets

3.1 Introduction

Petri nets (PN) are one of the most popular models of concurrent systems, used by both theoreticians and practitioners. They are a graphical and mathematical tool of parallel systems, in the same way that the finite automatons are a graphical and mathematical tool of sequential systems. PNs have been used to study systems that can be modeled at some level of abstraction as discrete-event dynamic systems. A Discrete Event System (DES), in contrast to Continuous Systems (CS) modeled as algebro-differential equations or qualitative abstractions, is defined as a dynamic system that evolves in accordance with abrupt occurrences, at possibly unknown, irregular intervals, of physical events [13].

The model-based diagnosis of DES has received a lot of consideration over the last decade being applied in various technological areas. Besides the "naturally discrete" systems, the quantization of the variables' change of the continuous and hybrid systems makes the discrete modeling possible.

The aim of this chapter consists to survey the use of PNs in model-based diagnosis. It starts in the following section by outlining some basic definitions (stated briefly since they are standard) about PNs. Section 3.3 considers the formulation of diagnosis problem within PNs. Resolution methods by analyzing PN models are presented in section 3.4. Section 3.5 discusses implementation architectures of DES diagnosis according to a topological point of view. The discussion considers implementations based on automata models as well as PN

ones. Finally, section 3.6 concludes the chapter.

3.2 Petri nets: Outline

This section outlines briefly some basic definitions on which we will rely throughout the rest of the thesis. An interested reader is referred to [60] for more details.

Definition 1 *A Petri net is a triple* $\mathcal{N} = \langle P, T, F \rangle$ *where*

- $P \cap T = \emptyset$
- $P \cup T \neq \emptyset$
- $F \subseteq (P \times T) \cup (T \times P)$
- $dom(F) \cup cod(F) \subseteq P \cup T$.

P is the set of places, T is the set of transitions and F is the flow relation represented by means of directed arcs. If the transitive closure F^+ of the arcs is irreflexive, the net is said to be *acyclic*. In a Petri net, an arc multiplicity function is usually defined as $W : (P \times T) \cup (T \times P) \longrightarrow \mathbb{N}$; if W is such that $W(f) = 1$ if $f \in F$ and $W(f) = 0$ if $f \notin F$, the net is said to be an *ordinary Petri net*. For each $x \in P \cup T$ we will use the classical notations ${}^\bullet x = \{y \mid yFx\}$ and $x^\bullet = \{y \mid xFy\}$. If ${}^\bullet x = \emptyset$, x is said to be a *source*; while if $x^\bullet = \emptyset$, x is said to be a *sink*. A marking is a function $\mu : P \longrightarrow \mathbb{N}$ from places to nonnegative integers represented by means of *tokens* into places. A marked Petri net is a pair $\langle \mathcal{N}, \mu \rangle$ where $\mathcal{N} = \langle P, T, F \rangle$ is a Petri net and μ is a marking.

The dynamics of the net is described by moving tokens from places to places according to the following definition of *enabling (i.e. concession)* and *firing rules*.

Definition 2 *Let* $\langle P, T, F, \mu \rangle$ *be a marked ordinary Petri net; a transition* $t \in T$ *is enabled at* μ *if and only if* $\forall p \in {}^\bullet t : \mu(p) \geq 1$*; if* t *is enabled at* μ*, then* t *may occur (fire) yielding a new marking* μ' *(we write* $\mu[t\rangle\mu'$*) such that for every place* $p \in P$ *we have:* $\mu'(p) = \mu(p) - W(p,t) + W(t,p)$.

The reachability set from a marking μ_0, indicated as $\mathcal{R}(\mathcal{N}, \mu_0)$ (or $[\mu_0\rangle$), is the smallest set of markings such that: 1) $\mu_0 \in \mathcal{R}(\mathcal{N}, \mu_0)$; 2) if $\mu_1 \in \mathcal{R}(\mathcal{N}, \mu_0)$ and $\mu_1[t\rangle\mu_2$ for some $t \in T$, then $\mu_2 \in \mathcal{R}(\mathcal{N}, \mu_0)$. If a place of a marked net cannot be marked with more than one token, the place is said to be *safe*; if the property holds for every place, the net itself and every marking are said to be *safe*. Moreover, let us recall the following definition of covering.

Definition 3 *Let $Q \subseteq P$, a marking μ of $\mathcal{N}$ covers Q if and only if $\forall p \in Q \rightarrow \mu(p) = 1$; while it zero-covers Q if and only if $\forall p \in Q \rightarrow \mu(p) = 0$.*

Given a Petri net $\mathcal{N} = \langle P, T, F\rangle$ with $n = |T|$ and $m = |P|$, the *incidence matrix* of $\mathcal{N}$ is the $n \times m$ matrix of integers $A = [a_{ij}]$ such that $a_{ij} = W(i,j) - W(j,i)(i \in T, j \in P)$. An m-vector of integers Y such that $A \cdot Y = 0$ is said to be a *P-invariant* of the net represented by A, the entry $Y(j)$ corresponds to place j. The support σ_Y of a P-invariant Y is the subset of places corresponding to nonzero entries of Y. In a dual way, if A^T is the transpose matrix of A, an n-vector of integers X such that $A^T \cdot X = 0$ is said to be a *T-invariant* (entries corresponding to transitions). It is well known that any invariant can be obtained as a linear combination of invariants having minimal (with respect to set inclusion) supports.

Definition 4 *Given a Petri net $\mathcal{N}$, $\wp = p_0 t_1 ... t_n p_n$ is a non-trivial path (or simply a path) in $\mathcal{N}$ if: i) $n > 0$; ii) $t_{q+1} \subseteq p_q^\bullet \cap {}^\bullet p_{q+1}$ for $q = 1, ..., n$.*

3.3 Modeling diagnosis with PNs

PN, as a tool for dynamic systems whose state changes with an event occurrence, can model the complicate man-made systems whose behaviors are hard to predict and offer the possibility to perform the automated fault diagnosis during the system execution.

In the context of DES diagnosis, the system to be diagnosed is modeled by a labeled Petri net $(\mathcal{N}, \Sigma, l, \mu_0)$ where Σ is the set of event labels for the transitions in T, $l : T \longrightarrow \Sigma$ is the transition labeling function, and μ_0 is the initial state. The event labeling function l is extended to $l : T^* \longrightarrow \Sigma^*$ in the following manner: given $t, t' \in T$ and $a, a' \in \Sigma$:

$$l(t) = a \text{ and } l(t') = a' \Rightarrow l(tt') = l(t)l(t') = aa'$$

The language generated by the labeled Petri net $(\mathcal{N}, \Sigma, , l, \mu_0)$, denoted by $\mathcal{L}(\mathcal{N}, \Sigma, , l, \mu_0)$, is the set of all traces of events that can be generated by $(N, \Sigma, , l, \mu_0)$ from its initial state μ_0. $\mathcal{L}(\mathcal{N}, \Sigma, , l, \mu_0)$ is formally defined as

$$\mathcal{L}(\mathcal{N}, \Sigma, , l, \mu_0) = \{l(s) \in \Sigma^* : s \in T^* \text{ and } \exists \mu : \mu_0[s\rangle\mu\}$$

Some of the events in Σ are observable, i.e. their occurrence can be observed (detected by sensors), and while the other events are unobservable. Thus Σ is partitioned into observable and unobservable event sets: $\Sigma = \Sigma_o \cup \Sigma_{uo}$. The observable events in the system may be commands issued by controllers, sensor readings, and changes of sensor readings. On the other hand, unobservable events may be some events that cause changes in the system state that are not recorded by sensors.

3.3.1 Fault representation

In some DES, faults set F can be alternatively represented as forbidden system states, faulty behavior modes of components, or unobservable events. In the case where faults are modeled as unobservable events, the set of fault events Σ_f is a subset of Σ_{uo}. Furthermore, the set of fault events is partitioned into disjoint sets where each set corresponds to a different fault type. The motivation for doing so is that it might not be necessary to detect uniquely every fault event, but only the occurrence of one among a subset (type) of fault events. We write

$$\Sigma_f = \Sigma_{F_1} \cup \cdots \cup \Sigma_{F_k}$$

where Σ_{F_i} denotes the set of fault events corresponding to a type i fault, $1 \leq i \leq k$, where k is the number of fault types. When we write "a fault of type i has occurred", we mean that a fault event from the set Σ_{F_i} has occurred.

Note that there are some other kinds of fault representation like violation of event execution conditions [44], violation of constraints on the target states, or logical propositions defined over a set of variables that comprise both events and states [42].

3.3.2 Observation

The observation of DES is retrieved from the monitoring system, which supervises the running of the system. Once it captures a symptom, the diagnosis process is triggered. The observation offered by the monitoring system is event-observation (as observation trace) and/or the partial state-observation (as symptom).

Assumption 1 *Unless otherwise stated, we make an important assumption: the fault cannot be in the monitoring components that log the information, which means the observation is accurate.*

For DES, the most common observation is the occurrence of events. In reality, the sensors or monitoring platform in charge of the observations can be malfunctioning. So the observation sequence can be inaccurate, incomplete, partially ordered, etc. In fact, many works effort to completely or partially release these assumptions to meet the real-life request from the industrial areas.

Observation absence

Due to the limitation of the observation, there could be an observation absence, e.g., some states or events occurrence are naturally hard or too expensive to capture. Then the diagnosis problem is explored in two directions: to improve the diagnosis confidence with available observation, or to carefully configure the sensors with higher diagnosis confidence and lower cost.

Partially ordered observation

An asynchronous system, much like an object-oriented software and a telecommunications network management system, is a system operating under distributed control, local time, global supervision, and components communication. Each local sensor has only a partial view of the system, and its local time is not synchronized with that of other sensors.

Even if the order of events may be correctly observed locally by each individual sensor, communicating alarm events via the network causes a loss of synchronization: as a result, the interleaving of events communicated to the supervisor is nondeterministic.

So the observation *Obs* has been defined formally as follows:

Definition 5 (Observable sequence) *Given an observable set Σ_o, a (partially ordered) observable sequence is defined as:*

$Obs ::= \varepsilon \mid event \prec Obs \mid Obs \parallel Obs$, *with* $event \in \Sigma_o$

with ε represents the empty observation, $\prec$ and $\parallel$ represent respectively the precedent and parallel relations between the events.

3.3.3 Diagnosis of DES

The PN diagnosis (or DES diagnosis) performs in two steps: deriving the legal traces which are consistent with the observation; then make the assertion [77]:

- if all the traces include a same fault transitions, the fault is declared to have happened for *sure*;
- if none of the legal traces include a fault event, the diagnosis result is *normal*;
- if the legal traces set includes traces that include different fault transitions and/or do not include fault transitions, the diagnosis result is *uncertain*.

So generally, the diagnosis of DES Δ_{DES} can be informally defined as follows:

Definition 6 *The diagnosis of a DES is a function* $\Delta_{DES} : traces_o(\mathcal{N}) \longrightarrow 2^{2^{F \cup \{N\}}}$, *with:*

- $traces_o(\mathcal{N})$ *is the set of observable traces;*
- F *is the set of fault types,* N *represents the normal state of the DES;*

3.4 Diagnosis methods

DES diagnosis methods are based on observing system events and making inferences about the system state. The basic idea is that the occurrence of a fault will generate a unique sequence of observable events that will establish the presence of the fault.

The classical diagnosis approach is to synchronize the system model for diagnosis and the observed traces for computing all compatible trajectories and determining whether these trajectories (a sequence of states and transitions) are normal. The system model for diagnosis can be represented as:

- synchronization product of DES model and fault types (Automata-diagnoser) [51, 77, 78, 85] and PN diagnoser [3];
- PN unfolding [8, 35] and backward unfolding [46].
- Petri net reachability graph [2, 11, 39, 56, 69, 86];

In this section, the above approaches are introduced and compared.

3.4.1 Diagnoser

Assume faults are represented as unobservable events and $F \subseteq \Sigma_{uo}$ is the fault types set. Given a labeled transition system (LTS), the composition product of the system states and the possible faults types can represent off-line the diagnosis states of the system, named as diagnoser [77, 78]. So a diagnosis can be got by synchronizing the diagnoser and an observable trace.

For PN models, the diagnoser is a labeled Petri net built from the system model $(\mathcal{N}, \Sigma, , l, \mu_0)$. This labeled Petri net performs diagnostics while observing on-line the behavior of $(\mathcal{N}, \Sigma, , l, \mu_0)$.

Definition 7 *The diagnoser for* $(\mathcal{N}, \Sigma, , l, \mu_0)$ *is* $\mathcal{N}_d = (\mathcal{N}, \Sigma, , l, \mu_{d_0}, \Delta_f)$ *where* $\mathcal{N}, \Sigma, l$ *are defined as before,* μ_{d_0} *is the initial diagnoser state and* $\Delta_f = \{F1, F_2, \cdots, F_k\}$ *is the finite set of fault types.*

The Petri net diagnoser $\mathcal{N}_d$ keeps the graphical structure of the underlying system model. Up to this point $\mathcal{N}_d$ is not different from a labeled Petri net. However, its dynamics are different from those of a labeled Petri net since its state transition function is only defined for observable events.

The diagnoser gives the estimate of the current state of the system after the occurrence of an observable event. The diagnoser state is a list of the set of states the system model can

be in after observation of an event in Σ_o together with fault information. Fault information in a diagnoser state is coded by fault labels.

Diagnosis with diagnosers is very efficient because processing an observation sequence can be done in linear time in the length of the sequence. However, the construction of the diagnoser may be extremely expensive because the diagnoser may have a size that is exponential in the number of states in the system, which is famous as the state-space explosion problem.

[77, 78] described a modeling and diagnosis framework based on FSMs for systems in the DES framework. A diagnoser based on the system model functions as an extended observer that provides estimates of the system state under non-faulty and faulty conditions.

[38] proposed a diagnoser approach by combining each marking with its exclusive diagnosis information, and got diagnosis by synchronizing the diagnoser with the observations. [38] requires the PN model to be more specific so that each marking corresponds either to a correct state or to one type of fault.

3.4.2 PN unfolding

Net unfolding is a technique of structural analysis to reduce the state-space explosion problem which the reachability analysis approaches suffer from. The unfolding of a system fully describes its concurrent behavior in a single branching structure, representing all the possible computation steps and their mutual dependencies, as well as all reachable states; the effectiveness of the approach lies in the use of partially ordered runs, rather than interleavings, to store and handle explanations extracted from the system model.

The unfolding definitions are taken from [8] and slightly adjusted.

Definition 8 (PN Homomorphism) *Given two PN graphs* $\mathcal{S} = \langle P, T, W \rangle$ *and* $\mathcal{S}' = \langle P', T', W' \rangle$ *a homomorphism from* $\mathcal{S}$ *to* $\mathcal{S}'$ *is defined as* $\varphi : P \cup T \longrightarrow P' \cup T'$ *s.t.,*

- $\varphi(P) \subseteq P'$ *and* $\varphi(T) \subseteq T'$
- $\forall x \in P \cup T, \varphi({}^\bullet x) = {}^\bullet\varphi(x)$ *and* $\varphi(x^\bullet) = \varphi(x)^\bullet$

Definition 9 (Occurrence net) *Given a PN graph $\mathcal{S} = \langle P, T, W \rangle$, two nodes x, x' are in conflict, noted as $x\#x'$, if $\exists t, t' \in T$, s.t. ${}^\bullet t \cap {}^\bullet t' \neq \emptyset$; and $t \preccurlyeq x, t' \preccurlyeq x'$ where $\preccurlyeq$ is a reflexive transitive closure of W. A node x is in self-conflict if $x\#x$. An occurrence net $\mathcal{O} = (B, E, \preceq)$ satisfies:*

- *B, a set of conditions;*
- *E, a set of transitions;*
- *$\preceq$ is the causality relation;*
- *$\forall x \in B \cup E : \neg[x\#x]$;*
- *$\forall x \in B \cup E : \neg[x \prec x]$ (acyclic);*
- *$\forall x \in B \cup E : |\{y : y \prec x\}| < \infty$ (well formed);*
- *$\forall b \in B : |{}^\bullet b| \leq 1$, each place has at most one input transition (no backward conflict).*

  We denote $min(\mathcal{O}) \subseteq B$ as the minimal[1] node set of $\mathcal{O}$ for $\preceq$.

Definition 10 (Cut) *Two nodes x, x' are concurrent, denoted as $x \perp x'$ if neither $x \preccurlyeq x'$, nor $x' \preccurlyeq x$, nor $x\#x'$. A maximum concurrent conditions (or pairwise nodes) set is a cut.*

Definition 11 (Configuration) *A configuration $\mathcal{C} = \langle B_\mathcal{C}, E_\mathcal{C}, \preccurlyeq_1 \rangle$ of $\mathcal{O}$ is defined as follows:*

- *$\mathcal{C} \subseteq \mathcal{O}$, $\mathcal{C}$ is a sub-net of $\mathcal{O}$;*
- *$\forall a, b \in (B_\mathcal{C} \times E_\mathcal{C}) \cup (E_\mathcal{C} \times B_\mathcal{O}) \Longrightarrow \neg(a\#b)$, $\mathcal{C}$ is conflict-free;*
- *$\forall b \in B_\mathcal{C} \cup E_\mathcal{C} : a \in B$ and $a \preccurlyeq_1 b \Longrightarrow a \in B_\mathcal{C} \cup E_\mathcal{C}$, $\mathcal{C}$ is up-warded closed;*
- *$min_\preccurlyeq(\mathcal{C}) = min_\preccurlyeq(\mathcal{O})$, $\mathcal{C}$ and $\mathcal{O}$ have the same starting nodes.*

We denote ζ as the configurations set of $\mathcal{O}$.

[1] $min_\preceq(X) = \{x \in X \mid (x' \in X \wedge x' \preceq x) \Rightarrow x' = x\}$ *is the minimal element of X.*

Definition 12 (Branching process) *Given a Petri net system $\mathcal{S}$, a branching process $\mathcal{B}$ is a pair $(\mathcal{O}, \varphi)$ where $\mathcal{O}$ is an occurrence net and φ is a homomorphism from $\mathcal{O}$ to $\mathcal{S}$, with:*

- $min(\mathcal{O}) = \mu_0 \Longrightarrow \varphi(min(\mathcal{O})) = \mu_0$
- $\forall e, e' \in E, {}^\bullet e = {}^\bullet e' \wedge \varphi(e) = \varphi(e') \Longrightarrow e = e'$

Definition 13 (Unfolding) *Given a Petri net system $\mathcal{S} = \langle \mathcal{N}, \mu_0 \rangle$, the unfolding $\mathcal{U}_\mathcal{N}(\mu_0)$ is a branching process $\mathcal{B} = (\mathcal{O}, \varphi)$ s.t. $\forall \mathcal{B}' = (\mathcal{O}', \varphi') \sqsubseteq \mathcal{B}$ where $\mathcal{B}'$ is a prefix of $\mathcal{B}$, $\exists$ a homomorphism $\phi : B' \longrightarrow B$, s.t. $\phi(min(\mathcal{B}')) = min(\mathcal{B})$ and $\varphi \circ \phi = \varphi'$.*

So $\mathcal{U}_\mathcal{N}(\mu_0)$ maximally unfolds $\mathcal{S}$ and configurations are the adequate representations of the firing sequences of $\mathcal{S}$.

So the diagnosis based on Petri net unfolding can be defined as:

Definition 14 (Diagnosis of PN unfolding) *Given a diagnosis problem $\langle \mathcal{U}_\mathcal{N}, Obs \rangle$ with $\mathcal{U}_\mathcal{N}$ the unfolding of a Petri net system $\mathcal{S} = \langle \mathcal{N}, \mu_0 \rangle$, the diagnosis is $Diag_{\mathcal{U}_{\mathcal{P}_\mathcal{N}}} = \{t \mid l(t) \in \Sigma_{uo}, \exists \tau$ is an observable trace of $\mathcal{N}$, s.t. τ is consistent with $\mathcal{U}_\mathcal{N} \times Obs\}$.*

[8] used a net unfolding approach for designing an on-line asynchronous diagnoser. The state explosion is avoided but the on-line computation can be high due to the on-line building of the PN structures by unfolding.

3.4.3 PN backward reachability analysis

Reachability analysis has been successively developed essentially by taking into account forward reachability. While backward reachability analysis is suitable for the diagnostic problem solving [2, 12, 43, 81]. The backward reachability analysis starts from the final marking which represents a symptom and calculates backwardly according to the backward searching rules to detect all the traces that cover it. So the backward calculation can be seen as a forward calculation in the reverse PN obtained by reversing the direction of the arcs in the original PN and modifying the enabling and firing rule of a transition.

[2, 69] proposed the backward reachability analysis (B-W analysis) approach on a particular class of PN called "*Behavioral Petri Nets*"(*BPNs*) to model the causal behavior of the

system to be diagnosed. The states of the system are represented as places, and the inferring relations between the states are represented as transitions. So the BPN model represents all the possible logical inferring paths of the system states. More formally, a BPN model has been defined as:

Definition 15 *A Behavioral Petri Net (BPN) is a 4-tuple* $\mathcal{N} = (P, T_N, T_{OR}, F)$ *such that* $(P, T_N \cup T_{OR}, F)$ *is an acyclic ordinary Petri net that satisfies the following axioms:*

1. $\forall p \in P(|{}^\bullet p| \leq 1 \wedge |p^\bullet| \leq 1)$
2. $\forall p_1, p_2 \in P({}^\bullet p_1 = {}^\bullet p_2) \wedge (p_1^\bullet = p_2^\bullet) \rightarrow (p_1 = p_2)$
3. $\forall t \in T_N(|{}^\bullet t| = 1 \wedge |t^\bullet| > 0) \vee (|{}^\bullet t| > 0 \wedge |t^\bullet| = 1)$
4. $\forall t \in T_{OR}(|{}^\bullet t| \geq 2 \wedge |t^\bullet| = 1)$

Such a net model is characterized by the following features:

- each place corresponds to an instance of the causal model states and transitions describe the cause-effect relationships between the corresponding instances of states;
- a source place (i.e. $p : {}^\bullet p = \emptyset$) corresponds necessarily to an initial state instance;
- a sink place represents either a manifestation state instance or an internal state instance that has no consequences;
- the net model is safe; that is, any place can be marked with at most one token;
- The set of transitions is partitioned into two subsets T_N and T_{OR}. Transitions in T_N (and-transitions) are intended in the usual way; while transitions in T_{OR} (Or-transitions) are intended to represent the logical connective OR and they represent macro transitions whose semantics can be given in terms of a Petri net with inhibitor arcs (figure 3-1). Informally, a transition $t \in T_{OR}$ (graphically represented as an empty thick bar) has concession in a marking if and only if at least one of its input places is marked. We refer to [69] for more details about formal definitions of enabling and firing rules of Or-transitions;

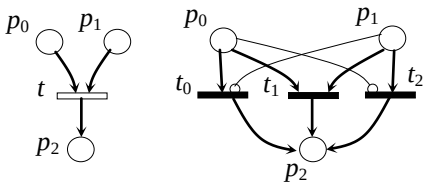

Figure 3-1: Or-transition and its semantics.

- an and-transition t is of type *linear*, *fork* or *join*. A transition t is a linear transition iff it has exactly one input place and one output place. A fork transition is that with only one input place and at least two output places. Finally, a join transition is the one with at least two input places and only one output place (for the sake of simplicity and to face the state explosion problem, a fork transition has two output places and a join transition has two input places. If $t \in T_{OR}$, then it has two input places and one output place);

- The transitive closure of the flow relation (i.e. arcs) is irreflexive[2];

- an initial marking of a BPN is a safe marking μ_0 such that if $\mu_0(p) = 1$ then p is a source place;

- a marked BPN is a pair $(\mathcal{N}, \mu)$ where $\mathcal{N}$ is a BPN and μ is either an initial marking or a marking such that there exists an initial marking μ_0 and $\mu \in \mathcal{R}(\mathcal{N}, \mu_0)$.

In this view, a BPN diagnostic problem $\mathcal{BPNDP}$ corresponding to a logical $\mathcal{DP}$ is defined as $\mathcal{BPNDP} = (\mathcal{N}, P^{ini}, \langle P^+, P^- \rangle)$, where $\mathcal{N}$ is the net corresponding to a causal BM, P^{ini} is a set of source places denoting initial states in BM, P^+ and P^- are two sets of sink places representing the observations and thus corresponding respectively to Ψ^+ and Ψ^-. Before showing how to characterize diagnostic solutions for a given $\mathcal{BPNDP}$, let us recall the following definition.

Definition 16 *a marking μ of a BPN such that no transition is enabled at μ is called a final marking.*

[2]This is a common assumption when modeling the causal behavior of a given system without taking into account temporal aspects.

The following theorem has been proved in [69].

Theorem 1 *In a marked BPN there is exactly one final marking.*

The notion of diagnostic solution can now be captured as follows:

Definition 17 *Given a* $\mathcal{BPNDP} = (\mathcal{N}, P^{ini}, \langle P^+, P^- \rangle)$*, an initial marking* μ^{ini} *is a solution to* $\mathcal{BPNDP}$ *if and only if the final marking* μ *of* $\mathcal{N}$ *covers* P^+ *and zero-covers* P^-*.*

BW-Analysis

An obvious consequence of the irreflexivity of the flow relation is that it defines a partial order, denoted as "$\prec$", over transitions of a BPN; given two transitions t_1 and t_2: $t_1 \prec t_2 \Leftrightarrow t_1 F^+ t_2$ where F^+ denotes the transitive closure of the flow relation. In order to implement diagnostic reasoning which is by nature hypothetico-deductive on a BPN model, a BW-Analysis has been defined in [2]. It consists in a backward reachability analysis accomplished with two types of tokens called respectively *normal* and *inhibitor* tokens. The meaning of normal tokens is as usual: we can associate a condition with a place and a normal token into a place means that such a condition is satisfied. On the contrary, inhibitor tokens represent conditions which are certainly not satisfied in the case under examination (i.e. for diagnostic purposes, they represent all parameter values that are different from the observed ones). If a place denoting a condition C is marked with this kind of token, then $\neg C$ holds. Consequently, when a place is empty, no constraint is imposed on the associated condition. As it is noted in [2, 69], this corresponds to considering a three-valued logic whose truth values are $\{true, false, unknown\}$.

Referring to the previous paragraph, in a causal BPN, a transition t is enabled, in forward fashion, at a marking μ (and so it can fire) iff t has concession in μ and $\nexists t' \prec t$ such that t' has concession in μ. This corresponds to imposing a priority ordering on transitions. Nevertheless, because we have to fire transitions in a backward fashion, the inverse relation of the partial order has to be considered: given two transitions t_1 and $t_2 : t_1 \succ t_2 \Leftrightarrow t_2 \prec t_1$.

b-w marking. The concept of b-w marking has been defined as a function μ from the set of places to the set $\{b, w, 0\}$; where $\mu(p) = b$ means that the place p is marked with a normal

token (black); $\mu(p) = w$ means that the place p is marked with an inhibitor token (white) and $\mu(p) = 0$ means that p is empty.

A particular feature of the BW-Analysis is the possibility of forcing the backward firing of and-transitions. Informally, it means that if a fork transition t has at least one marked output place then all of its output places that are empty should be marked with the same type of token.

According to the logical definition of a diagnosis problem and of a solution to such a problem, we require a set of firing rules that will be applied in a backward fashion on the net model. Such rules start from a b-w marking corresponding to the observed misbehavior (a marking μ s.t $\mu(p) \neq 0 \Rightarrow p \in P^+ \cup P^-$) and ending up within an initial b-w marking in which the marked places belong necessarily to P^{ini} (i.e. corresponding to the initial states of the behavioral model). Such initial b-w marking represents the solution to the given diagnosis problem. In this case, empty source places represent initial conditions that are not significant for the case under examination, while source places marked with normal and inhibitor tokens represent initial conditions that have been proven true and false respectively.

In order to obtain such initial b-w marking, we need to construct the markings graph in a backward fashion by applying the set of firing rules that are defined in [2]. Figure 3-2 gives graphically these rules.

[46] adapted the PN unfolding method for backward searching. The set of minimal explanations is calculated backwards starting from the observation and deriving traces that lead back to the initial marking. The diagnoser explores different state spaces but has the advantage that it does not depend on the size of the PN model but only on the size of the largest sub-net in the model that includes only unobservable transitions. Moreover and very important the set of complete explanations can be calculated from the set of minimal explanations whenever this is required.

[39] studied the minimal diagnosis of unobservable-transitions-acyclic PN. A diagnosis approach named as basic reachability tree is proposed which is in fact an automaton diagnosis based on marking graph of Petri net. [11] studied the reachability graph diagnosis approach based on bounded PN model. The observations are transferred to a justification-vector to improve the efficiency.

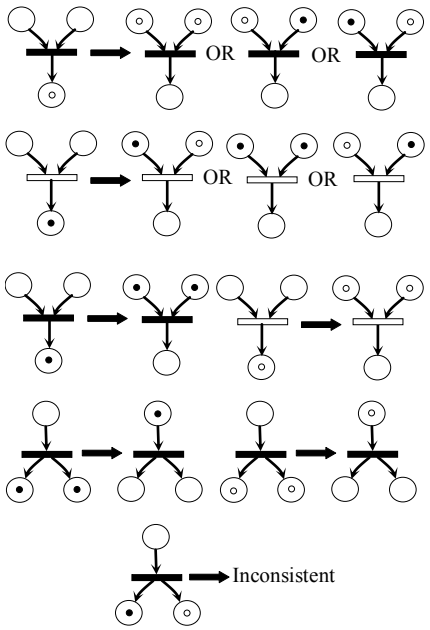

Figure 3-2: Backward firing rules.

[86] introduced a method for modification of reachability trees in order to detect failure transitions. A symbol ϖ means an infinite set of positive integers, so an infinite tree consisting of infinite reachable markings is approximated by a finite tree (reachability tree). Two kinds of diagnosers (difference marking ϖ-diagnoser and refined ϖ- diagnoser) were proposed. For observable places whose token numbers are replaced by ϖ in the reachability trees, the former diagnoser calculates difference between token numbers before and after partially observed markings change, and detects failures. In the latter diagnoser is refined to distinguish the reachable markings by normal and faulty behaviors.

[44] used PN models to introduce redundancy into the system and additional P-invariants allow the detection and isolation of faulty markings.

3.5 Architecture of DES diagnosis

The large DES systems are usually designed as a set of interconnected subsystems with different topological architecture, which can be roughly divided as decentralized and distributed ones. So decentralized and distributed diagnostic protocols become necessary to deal with diagnosis in distributed systems where the information is separately located.

The model-based diagnosis of DES can be classified in the literature from a topological point of view as centralized, decentralized, and distributed approaches.

3.5.1 Centralized diagnosis

There is one centralized diagnoser that derives the system diagnosis based on its (complete) knowledge of the overall system model and the overall system observation. The centralized approach can be further classified as:

- diagnoser approach [77] where a diagnoser automaton is derived off line and the on-line analysis is carried out by eliminating the diagnoser-states that are not consistent with the system observation.
- active system approach [3] where the diagnosis result is derived a posteriori when the system is in a quiescent state (out of work or idle).

The main disadvantage of a centralized approach is its high computational complexity. It requires a centralized model and generates a centralized diagnoser. Since the diagnoser-automaton can be viewed as a special observer-automaton its size may become too large to be practically stored. Even if a centralized diagnoser can be constructed it has the following disadvantages [82]:

- weak robustness: when the centralized diagnoser is broke down, the whole system is not able to be diagnosed.
- low maintainability: a change in the system structure requires a complete re-calculation of a new centralized diagnoser, which can be a serious problem for the dynamic systems.

3.5.2 Decentralized diagnosis

The decentralized diagnosis problem is first considered in [25] in which the local diagnosers communicate with the coordinator through the no-delay channels in stead of with each other. There is one coordinating agent receives information from several local diagnosers, each of which performs some local diagnosis of the system with incomplete knowledge (e.g. based

on a sub-set of sensor readings or a partial knowledge of the overall model). The local diagnosis results are compiled in a consistent diagnosis result for the overall system by the coordinating diagnoser e.g., [10, 25, 26, 62, 63].

There are two different decentralization levels:

- [25], or its extended version [26, 51], employ a global system model which is built from component models (given as FSMs) automatically via synchronous or asynchronous composition. After offline diagnosability verification, which may cause state explosion problem, the online diagnosis decisions can be computed. These decisions may or may not be fused on a coordinating site, according to the properties of the architecture. Three coordination protocols are proposed in [25] that realize the proposed architecture and analyze the diagnostic properties of these protocols.

- [3, 62, 64] and [45] (synchronous automata) proposed the decentralized system model as asynchronous communicating automata (or FSMs). [3] solved off-line a diagnosis problem a posteriori, while [52] mixed a diagnoser approach [51, 62] with an extended version of the decentralized model of [3] by computing on-line only the interesting parts of a centralized diagnoser to avoid computing the global model. [64] introduced the temporal window to improve the on-line diagnosis efficiency and the global diagnosis is built by dynamically merging the local ones to eliminate the inconsistent traces with the partial order reduction technique and incremental diagnosis on sound temporal windows. Recent works on this approach [22] used decentralized or factored representations to represent the set of all trajectories more compactly without enumerating all of them.

While decentralized models could potentially reduce the state space exponentially, the actual complexity of the diagnosis algorithms relies on the partition of the system model and the selection of communicating events between local models.

[75] investigated the necessariness of asynchronous communication for fault diagnosis. It modeled the asynchronous communications between two local diagnosers with timed automata. Then the problem of determining the states of each of the two communicating

diagnosers into the problems of factorization of the observation map and construction of an observer for a timed DES. The diagnosers can be formulated directly from the observers.

[10] studied the problem of synthesizing communication protocols and failure diagnosis algorithms for decentralized failure diagnosis of DES with costly communication between diagnosers. The costs on the communication channels may be described in terms of bits and complexity. The costs of communication and computation force the trade-off between the control objective of failure diagnosis and that of minimization of the costs of communication and computation.

[54] proposed a modular diagnosis architecture (broker) capable of merging diagnoses provided by local diagnosers and to enrich their formalism with synchronization constraints. The global diagnoser algorithm manages a diagnosis tree by querying the local diagnosers to complete the pending paths. Each candidate diagnosis is represented by a path leading to a constraintless node in the diagnosis tree.

The decentralized approaches overcome the high complexity and the low maintainability limitations of the centralized approach by calculating local state spaces (of size a lot smaller than the size of the overall) that are maintained consistent by a centralized structure (agent). But the existence of a centralized agent does not eliminate the disadvantage of a weak robustness.

[70] introduced a notion codiagnosability to describe a requirement that any failure can be diagnosed within bounded delay by at least one local diagnoser using it's own observations of the system execution. The codiagnosability property is stronger than diagnosability under the aggregate observations, which declaimed a possibility that a system is centrally diagnosable but not decentrally diagnosable.

[49, 50] focused on solving the ambiguity of several local diagnosis towards the global diagnosis (introduced and discussed in [87] and its extended version [88]). [50] proposed a framework for performing diagnosis in a decentralized setting. A global diagnosis decision is taken to be a winning local diagnosis decision, which is tagged with a certain ambiguity level. The work showed that the codiagnosability introduced in [70] was the same as 0-inference-diagnosability; the conditional codiagnosability introduced in [87] and [88] was a type of 1-inference-diagnosability; and the class of higher-index inference-diagnosable systems strictly

subsumed the class of lower-index ones. The authors of [88] claimed their architecture can be realized in a distributed environment.

3.5.3 Distributed diagnosis

In a distributed diagnosis environment, the overall system consists of different components, and associated with each component, there is a local agent (diagnoser-agent) that derives the local diagnosis of its component. Each local agent only knows the model of the local component and of its interactions with its neighbors. Each local agent moreover only receives signals from the monitoring system for the local events. No centralized structure is assumed to coordinate the results of the local agents but from time to time the local agents may exchange messages over communication channels linking them. Thus the local agents derive the distributed diagnosis by local calculations and by information exchanges, e.g., [35, 36, 38, 43, 46, 47, 72, 82, 83].

Generally speaking, distributed architectures for diagnosis differ from decentralized ones in terms of the local models used at the different sites for model-based inferencing and in terms of the ability for local diagnosers to communicate among each other in real-time.

For a distributed system without coordinator, the consistency check between the local diagnoses is merely important. The local consistency requires that all local diagnoses agree on their mutual interfaces. While the global consistency ([84]) requires the local diagnoses are the projected versions of the global diagnosis, which needs a global system model.

[71], extended from [70] defined a new observation mask for each local observer that combined the effect of it's own observation and the bounded-delay communication received by other diagnosers. The local diagnosers communicate with each other using the immediate observation passing protocol. Thus the distributed diagnosis problem is reduced to a decentralized diagnosis one. The further extended work [72] reduce the complexity of on-line diagnosis at each local site to be linear with the number of sites by proposing a new distributed diagnosis protocol.

[82, 83, 84] proposed an automaton-based distributed and hierarchical diagnosis architecture. Each local component has its own local diagnoser, which is built based only on knowledge about this component. The stored size of the overall diagnoser is only the sum

of state sizes of the local diagnosers, hence spatial complexity is kept under control. Each local diagnoser is connected with other local diagnosers based on the input/output relations among associated local components. Adding new components, taking components out of the system or changing the input/output relations among local components only affects the local diagnosers that are directly associated with the altered components. A hierarchical computational procedure and multi-resolution diagnosis approach are introduced in [84] to overcome the shortcomings of high time complexity and poor scalability of the distributed ones.

[1, 34, 35, 36, 37] discussed the distributed diagnosis problem based on PN model.

[35, 36] based on the work of [1, 8] discussed the distributed monitoring and diagnosis problems of the asynchronous subsystems with partial ordered observations. The idea here is that when concurrent subsystems are composed, there may be events in the alphabets of the subsystems whose relative order is not important. Therefore, partial-order techniques reduce the complexity of a model by not capturing all the permutations of the orderings of these events. [36] proposed and discussed different kinds of data structures (execution tree, unfolding, trellis, etc.) of representing the asynchronous communication, real concurrency and partial ordered events for the distributed diagnosis.

[34] modeled a distributed system as a graph of interacting subsystems, with the appropriate semantics of trajectories and stochastic framework. A centralized supervisor, collecting all observation from the system and knowing a model of the whole system, may not be affordable, so they advocate instead a processing by parts, and extend the idea towards a completely distributed supervisor architecture, with one local supervisor on the top of each subsystem, coordinating its activity with the supervisors in its neighborhood.

In [38], distributed diagnosis for Petri nets with synchronous communication is studied. The authors extend the notion of FSM diagnosers to PN and centralized and distributed diagnosers are designed. The centralized approach presents the same problems of combinational explosion than the original based on FSM and the distributed approach focuses on the problem of communication between the diagnosers.

[43, 47] proposed a distributed diagnosis based on place-bordered PN models, which are bounded, ordinary and known initial marking. A fault in a PN model is represented by

a choice transition. The case of unobservable interactions between components and cyclic communications are considered. The minimal explanations are derived by backward inference on each local diagnoser based on the local partial ordered observations. [47] concerned the diagnosis of plant systems, so the system model is assumed to be global clock scheduled instead of event-driven. A local diagnoser first searches for the minimal configuration to decide the initial marking with backward unfolding approach then infers forwardly for the possible local exited tokens to update the state of its neighbor diagnosers. The global consistency is verified by comparing the causal relations of the communicating events between the different sites with the observations. The state explosion problem is partially controlled with partial observation.

3.6 Conclusion

The correctness and efficiency of DES diagnosis depends mainly on three elements: the system model, the fault representation, and the diagnosis approach. On the level of system model, automata are suitable for monotonic system with smaller states set and larger events set; PN are suitable for real concurrent system. On the level of fault representation, faulty states and events are normally adopted separately. On the level of diagnosis approaches, the diagnoser, unfolding, and (backward) reachability approaches all suffer a lot from synchronization. The PN algebraic approach can help to improve the efficiency while the fault representation becomes difficult and complicated.

While there is no absolute adequate standard, the final choice depends on the system characteristics and the aim of diagnosis.

Part II

Contributions

Chapter 4

A distributed model-based diagnosis

4.1 Introduction

As it has been outlined, for large distributed systems the adoption of distributed multiple diagnostic agents can offer a solution to problems encountered by a single centralized diagnosis approach. Following such a direction, the system to be diagnosed is defined as a collection of interacting subsystems that may be geographically distributed. With each subsystem, we associate a diagnostic agent that knows the local model of the subsystem, receives the local observation, and can exchange limited information with the adjacent agents for consistency checking. Figure 4-1 displays a typical architecture for such a setting.

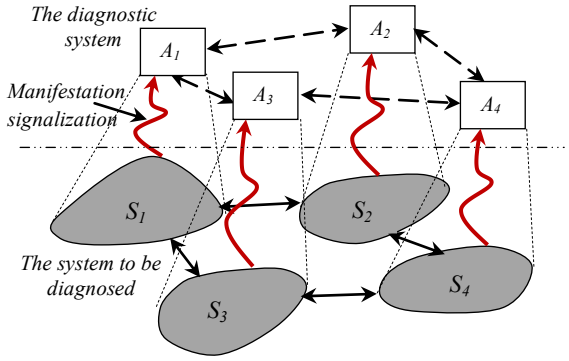

Figure 4-1: A diagnostic system architecture.

This chapter begins in the following section by discussing our logical view to a distributed model-based diagnosis problem, then in section 4.3 we characterize declaratively the diag-

noses of each agent in the diagnosis system. Section 4.4 tries to add more constraints to the diagnoses of each agent in order to recover the results that would be derived by a centralized agent having a global view about the whole system. Finally, section 4.5 concludes the chapter.

4.2 Problem statement

According to figure 4-1, we have a distribution of the knowledge over multiple agents. Such a distribution defines a division of a system into several subsystems. When knowledge is spatially distributed, the set of components $COMPS$ is partitioned over the agents. So, agent A_i has knowledge about components $COMPS_i$, and $COMPS = \bigcup_{i=1}^{n} COMPS_i$ where n is the number of agents. In other words, the initial diagnostic problem DP is divided into n sub-problems. Thus, $DP = \bigcup_{i=1}^{n} DP_i$. Each of these DP_i corresponds to a particular subsystem and can be viewed similar to the initial overall diagnostic problem DP. Nevertheless, each subsystem is not fully independent from the others and it interacts within other system parts through different connection points. Thus, by distributing knowledge, i.e. SD_i over the agents, we loose the knowledge about the connections between components managed by different agents. Hence, each sub-problem should include in its definition the connection elements relaying the corresponding subsystem within neighboring ones (i.e. we provide each agent with information about connection points that connect to components managed by other agents). More particularly, we split the set of connection points into relative inputs In_i and outputs Out_i of the agent's subsystem. Since we are concerned with diagnosis based on causal models, the discussion will be oriented toward such models.

Hence, $DP_i = (\langle BM_i, COMPS_i \rangle, Ctx_i, In_i, Out_i, \langle \Psi_i^+, \Psi_i^- \rangle)$ is a sub-problem to be solved by the agent A_i, where BM_i denotes the behavioral model of the subsystem S_i. Ctx_i is the set of possible fault causes originating from components of S_i[1]. In_i and Out_i correspond to connection points that are classified respectively into inputs to S_i that are determined from other subsystems S_j and outputs from S_i to S_j. $\langle \Psi_i^+, \Psi_i^- \rangle$ represent local observation

[1]If the BM_i describes the structure and the correct behavior of the subsystem S_i, then Ctx_i corresponds to the local context of S_i.

findings defined within S_i and their meanings are the same as for the centralized case. In this view, the diagnostic system can be considered as a multiple diagnostic agents, each of which is responsible to solve a local diagnostic problem DP_i and should communicate with neighboring agents for eliminating local solutions that are not consistent with results obtained by the neighborhood. Consequently, In_i can be viewed as initial states when BM_i models the causal behavior of S_i, since their causes originate from the outside of the associated model (i.e. their causes belong to other subsystems), and hence they are not explained locally. Similarly, elements of Out_i may be considered as manifestations to be explained even if they are unobservable; because they correspond to states that represent causes of other effects which are not modeled in BM_i. In fact, the consequences of elements of Out_i belong to the neighboring models.

4.3 The diagnosis of one agent

Each agent A_i in the multi-agent system must make a diagnosis of the subsystem S_i ($DP_i = (\langle BM_i, COMPS_i\rangle, Ctx_i, In_i, Out_i, \langle \Psi_i^+, \Psi_i^- \rangle)$). This can be viewed as a single agent diagnosis if values of the inputs and outputs of the subsystem are known. We use the set V_i to denote value assignments $value(p) = v$ with $p \in In_i$ to the inputs. V_i is the local context of the subsystem S_i that is determined by the outputs of other subsystems. We therefore extend Definition 7 given in chapter 2 to diagnosis of subsystems.

Definition 1 *Let $DP_i = (\langle BM_i, COMPS_i\rangle, Ctx_i, In_i, Out_i, \langle \Psi_i^+, \Psi_i^- \rangle)$ be a sub-problem to be solved by A_i and let V_i be a (partial) description of the values of connection points In_i. Finally, let Δ_i be a candidate diagnosis. Then,*

Δ_i is a diagnosis for S_i iff Δ_i is a diagnosis for $(\langle BM_i, COMPS_i\rangle, Ctx_i \cup V_i, \langle \Psi_i^+, \Psi_i^- \rangle)$.

This definition implies that a local diagnosis for DP_i can be regarded as a set of assumptions $\Delta_i \subseteq Ctx_i \cup V_i$ about the presence of some local faults such that:

$\forall m \in \Psi_i^+ : BM_i \cup V_i \cup \Delta_i \vdash m$

$\forall n \in \Psi_i^- : BM_i \cup V_i \cup \Delta_i \nvdash n$

In fact, diagnoses are to be given in terms of Ctx_i and parts relative to values of In_i (i.e. V_i) will be used later during communication between diagnostic agents. This is due to the fact that values of In_i represent inputs to S_i which correspond to the outputs of the neighboring subsystems that interact with S_i.

4.4 The diagnosis of multiple agents

Given multiple diagnostic agents, an important question is how the diagnoses of the agents relate to the diagnoses of a single agent that has complete knowledge of the system description and the observations. When addressing this question we assume throughout the thesis that there are no conflicts between the knowledge of the different agents. That is, there always exists a diagnosis Δ such that $\Delta \cup \langle BM, COMPS \rangle \cup Ctx \cup Obs$ is consistent. The following propositions have been proposed in [76].

Proposition 1 *Let $S_1, ..., S_k$ be the subsystems that make up the system S. Moreover, let Δ be a single agent diagnosis of S.*

Then $V_i = \{(value(p) = v) \mid p \in In_i, \Delta \cup \langle BM, COMPS \rangle \cup Ctx \vdash (value(p) = v)\}$ is the local context of S_i that is determined by the other subsystems S_j , and $\Delta_i = \Pi_{COMPS_i}(\Delta)$ is a diagnosis of S_i, where $\Pi_{COMPS_i}(\Delta)$ denotes the projection of Δ on $COMPS_i$.

Proposition 2 *Let $S_1, ..., S_k$ be the subsystems that make up the system S. Moreover, let the local context V_i of S_i describe the values of connection points in In_i that must be determined by the other subsystems S_j, and let Δ_i be a diagnosis of S_i determined by agent A_i given V_i.*

Then, $\Delta = \bigcup_{i=1}^{k} \Delta_i$ is a single-agent diagnosis if

1. *Δ is a candidate diagnosis,*
2. *$\Delta_i \cup \langle BM_i, COMPS_i \rangle \cup Ctx \cup V_i \vdash (value(p) = v)$ and*
3. *for every $p \in Out_i, p \in In_j : (value(p) = v) \in V_j$.*

The above propositions show that, in principle, multi-agent diagnosis is possible.

Complexity

If knowledge is spatially distributed, each agent manages a different part of the system. The behavior of a subsystem managed by an agent depends on the behavior of the other subsystems. This makes it difficult to predict the behavior of the whole system. The values of the connection points in Out_i depend on the local context V_i. The values specified by V_i, however, are determined by other subsystems S_j whose local context V_j may depend on the values of the connection points in Out_i. Because of these circular dependencies, predicting the systems behavior becomes an NP-Hard problem as it has been discussed in [76].

To avoid solving such a hard problem in the case where each SD_i models the structure and the correct behavior of the corresponding subsystem, consistency based diagnosis and consistency based diagnosis with abductive explanation of normal observations are preferred. When local models (i.e. SD_i) correspond to faulty causal behaviors obtained by partitioning a global causal model, then we are sure that such circular dependencies are inexistent because the initial overall causal model is by definition acyclic. We will return to such an assumption in the next two chapters.

Distributing the diagnostic process

After observing abnormal behavior of the system, the agents must make a diagnosis. In order to do so, each agent must make a local diagnosis in which it also takes into consideration the correctness of those inputs of its subsystem that are determined by other agents. Therefore, we must extend a candidate diagnosis Δ_i of agent A_i with correctness assumptions Ca_i about the systems inputs. For every input $p \in In_i$, Ca_i contains either the proposition $correct(p)$ or $\neg correct(p)$. The conditional context Cc_i will be used to describe inputs of a subsystem S_i, i.e. the local context of the subsystem determined by other subsystems, conditional to these correctness assumptions, i.e. $Cc_i = \{correct(p) \leftrightarrow (value(p) = v) | value(p) \in V_i\}$.

If in its local diagnosis (Δ_i, Ca_i), agent A_i assumes that one of its inputs is incorrect, the agent must communicate this information to an agent A_j determining the input. Next, agent A_j may treat this information as an observation of one of its outputs, and adapt its local diagnosis accordingly.

A problem with this approach is the occurrence of loops. Suppose that agent A_i blames

an observed anomaly on one of its inputs determined by the subsystem of an agent A_j. Agent A_j may also blame the fault in the output determining the input of S_i on one of its inputs. If this input is determined by an output of the subsystem of agent A_i, we may have a cycle of blames that supports itself. Clearly, a local diagnosis that constitutes such cycles of blames does not represent a valid diagnosis of the system. Moreover, handling such loops is a non trivial task which requires tracking dependencies between components in local diagnoses. In fact, the handling of loops causes the determination of minimal diagnoses to be an NP- hard problem.

For causal models, once Δ_i are obtained, they will be used to deduce instances of states corresponding to outputs of S_i. Such instances are to be compared for consistency checking with values requested by neighboring agents as their inputs. Furthermore, since Out_i may be viewed similar to the observable findings, they are classified also into two subsets Out_i^+ and Out_i^-. Out_i^+ corresponds to the output values that are modeled in BM_i and are deduced from Δ_i; whereas Out_i^- corresponds to the modeled values that contradict the deduced ones.

In logical terms:

$\forall a \in Out_i^+ : BM_i \cup In_i \cup \Delta_i \vdash a$

$\forall b \in Out_i^- : BM_i \cup In_i \cup \Delta_i \nvdash b$

Hence, Δ_i is considered as a local diagnosis which is consistent with diagnoses of the other agents iff:

$\forall m \in \Psi_i^+ \cup Out_i^+ : BM_i \cup In_i \cup \Delta_i \vdash m$

$\forall n \in \Psi_i^- \cup Out_i^- : BM_i \cup In_i \cup \Delta_i \nvdash n$

4.5 Conclusion

The discussion presented in this chapter characterizes distributed model-based diagnosis with an emphasize on declarative definitions. We will exploit such definitions in the case where the model of each subsystem captures its causal faulty behavior in the remainder of this thesis.

Notice that there has been other recent works that deal with the problem of distributed

causal model-based diagnosis. We mention in this regard the work presented in [9] in which the authors concentrate on how to partition a global model of the system to be diagnosed to a set of local models (regions in the terminology of [9]). They propose a framework ensuring that the obtained local regions are maximally independent and their common elements are minimal. The maximal independency level allows diagnosis to be performed locally, without the need of communicating with the other models, as long as the border with them is healthy; and the minimum number of borders ensures that the communication among diagnostic agents is minimal.

Chapter 5

A distributed BW analysis

5.1 Introduction

This chapter considers the problem of spatially distributed causal model-based diagnosis. The system to be diagnosed is viewed as a collection of interacting subsystems in which when a fault occurs in one subsystem, it may generate some fault indications (i.e. symptoms) and may propagate to the neighborhood. The diagnostic system itself reflects a similar structure of the system to be diagnosed and is defined as a set of diagnostic agents each of which is associated with a specific subsystem. In particular, each agent has a local model, given as a behavioral Petri net (BPN) model, of the assigned subsystem and may receive observations generated only by elements of this subsystem. The local BPN model describes the causal behavior of the subsystem as well as its interactions within adjacent ones. These interactions are captured through tokens that may pass via common bordered places between BPNs. When agents observe an aberrant behavior (modeled as the marking of some sink places in the local BPN models), each one is charged to explain (or to diagnose) the received local observation on the basis of its BPN model. This is accomplished locally by a backward analysis (BW-Analysis) of the corresponding reachability graph. As a result, each agent obtains a set of local initial markings from which diagnoses have to be given.

In order to achieve the consistency with the local diagnoses of all other agents, each one requests from its neighbors the required marking of its bordered places for each computed diagnosis. At this step, agents receiving such a request will construct their reachability

graphs in a forward fashion to check if the requested marking of bordered places is reachable from at least one of their computed initial markings. If so, the local diagnosis from which the exchanged message has been generated is considered globally consistent; otherwise, it is not supported by diagnoses of the neighborhood and consequently it must be discarded.

The remainder of this chapter is organized as follows. In Section 5.2, we start with a description of the system model as a set of bordered places BPNs. In Section 5.3, we state the problem of distributed causal model-based diagnosis on BPN models. The local diagnosis algorithm is based on constructing backwardly the corresponding reachability graph. The last part of the section describes the cooperation protocol used by diagnostic agents to verify global consistency between local solutions. Finally, in Section 5.4, we give some concluding remarks.

5.2 The system model

In terms of BPNs models, each local diagnosis problem $\mathcal{DP}_i$ is represented as a BPN diagnostic problem with bordered places. In fact, the idea of representing interactions between different subsystems through tokens that pass (either observably or unobservably) via bordered places, also called common places, in Petri net models has been exploited by some researchers [38, 43, 46]. In these works, the passing of tokens is considered observable if transitions that produce them are observable since transitions are labeled by events. When the labels correspond to unobservable events, the token passing is considered as unobservable. According to the logical definition given in the previous chapter, connection elements between subsystems are not necessarily unobservable. Thus, when such an element is observable, we mean that the corresponding interaction may be observed by the diagnostic agents. Consequently, the BPN diagnostic problem can be defined as: $\mathcal{BPNDP} = \bigcup_{i=1}^{n} \mathcal{BPNDP}_i$. Each $\mathcal{BPNDP}_i$ corresponds to $\mathcal{DP}_i$, and is defined as:

$\mathcal{BPNDP}_i = (N_i, P_i^{In}, P_i^{Out}, \langle P_i^+, P_i^- \rangle)$, where:

- $N_i = \langle P_i, T_i, F_i \rangle$ is a behavioral Petri net corresponding to BM_i;
- P_i^{In}, P_i^{Out} are sets of places denoting the elements of In_i and Out_i respectively;

- and $\langle P_i^+, P_i^- \rangle$ are places that represent the observed manifestations of the subsystem S_i.

Thus, the set of BPNs $\{N_i \mid i = 1...n\}$ can be viewed as a partition of a global net model $N = \langle P, T, F \rangle = \bigcup_{i=1}^{n} N_i$ such that:

1. $P = \bigcup_{i=1}^{n} P_i$, and $\forall i \Rightarrow \exists j$ s.t $P_i \cap P_j \triangleq P_{ij} \neq \emptyset, P_{ij} \subseteq P_i^{In} \cup P_i^{Out}$;
2. $T = \bigcup_{i=1}^{n} T_i$, and $\forall i \neq j \Rightarrow T_i \cap T_j = \emptyset$;
3. $P_i^{In} = \{p \mid (p^\bullet \in T_i) \wedge ({}^\bullet p \notin T_i)\}$;
4. $P_i^{Out} = \{p \mid ({}^\bullet p \in T_i) \wedge (p^\bullet \notin T_i)\}$.

In order for N to be a causal behavioral model, it will be, according to the definition given in section 3.4.3, acyclic. The acyclicity of the global net N may be violated by the presence of a cycle in at least one of the local net models and/or in the interactions among local models. Since each local BPN is acyclic by definition, we will assume that the interactions between local models is also acyclic. Such interactions can be described by a directed graph $IG = (V, E)$ (IG for Interaction Graph), whose nodes (elements of V) correspond to common bordered places between local BPNs. An arc $(p, p') \in E$ iff $\exists$ a path $\wp$ joining p to p' in a local BPN. Thus, we make the following assumption.

Assumption 1 *The graph IG representing interactions between all local BPNs is acyclic.*

In fact, when such assumption is relaxed, the diagnosis problem becomes an NP-hard as it has been discussed by [76] in the context of logical frameworks, because an agent A_i, after observing a malfunction of the subsystem S_i, may blames a neighboring subsystem S_j managed by A_j; similarly A_j may blames S_i to explain such observation. As a result, we may have a cycle of blames that supports itself.

Example 5.1. As an example, let us consider a system S composed of two interacting subsystems S_1 and S_2. The model of each subsystem is described by a BPN representing its faulty causal model. Figure 5-1 gives the graphical representation of the corresponding models. Dotted circles, labeled $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{z}$, represent the common places that are

used to model interaction between the two subsystems. $\boldsymbol{x}$ and $\boldsymbol{z}$ model the fact that tokens can pass from BPN_2 to BPN_1; while $\boldsymbol{y}$ models the inverse direction. The models are adapted from an example given in [67] which is used to represent a partial fault centralized model of a car engine. BPN_1 is characterized by the entities $pist\_ring\_state(worn)$, $pist\_state(worn)$ and $oil\_sump\_state(worn)$ as local initial states of the described causal model, and $ex\_smoke(black)$, $oil\_light(red)$ and $accel\_resp(del)$ as local manifestations. Similarly, for BPN_2, three local initial states are considered, they are modeled by places $road\_cond(poor)$, $ground\_clear(low)$ and $spark\_plug\_meleage(high)$, and two local manifestations $hole\_oil\_sump(yes)$ and $temp\_ind(red)$. Transitions of each net model the cause-effect relationships among the corresponding entities; for example in BPN_1, transition t_1 models the fact that an "increased oil consumption"(modeled by place $oil\_cons(incr)$) is caused by both a "worn state of piston rings"(modeled by place $pist\_ring\_state(worn)$) and a "worn state of pistons"(modeled by place $pist\_state(worn)$). In our discussion, the meaning of the different modeled entities is irrelevant, since our aim is to show how to implement diagnostic inference reasoning by analyzing such net models.

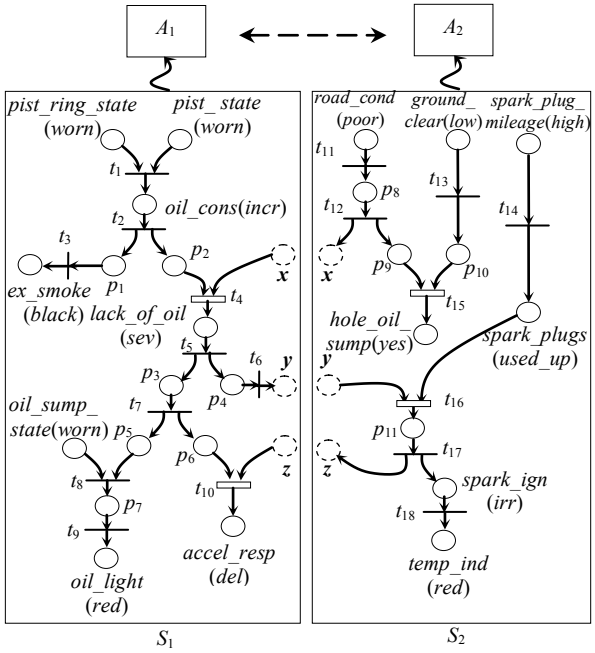

Figure 5-1: Example of a distributed BPN.

5.3 A distributed diagnostic reasoning scheme

After receiving a local observation, each diagnostic agent in the multi-agent diagnostic system starts to explain locally such an observation, and then begins to communicate within its neighbors to recover the results of a centralized agent having a global view of the whole system.

5.3.1 A distributed BW-Analysis

For explaining locally the received observation, agent A_i tries to construct, in a backward fashion, the reachability graph of its local model. This is accomplished by applying the set of backward firing rules, discussed in section 3.4.3, on the local BPN from a final marking $\mu_i^{Obs_i}$ corresponding to the received observation Obs_i (i.e. $\mu_i^{Obs_i}$ s.t. $\mu_i^{Obs_i}(p) \neq 0 \Rightarrow p \in P_i^+ \cup P_i^-$). This intuitively corresponds to search in the local BPN_i from $\mu_i^{Obs_i}$ for an initial marking μ_i^{ini} which entails the observation marking ($\mu_i^{ini} \vdash \mu_i^{Obs_i}$). In Petri net notations, each agent A_i should calculate an initial marking μ_i^{ini} from the observation marking μ_i^{Obs}:

$$\begin{aligned} \mu_i^{ini} \quad = \quad & \{\mu \mid \forall p \in P_i, \mu(p) \neq 0 \Rightarrow {}^\bullet p = \emptyset \wedge \mu_i^{Obs} \in [\mu\rangle \texttt{ and} \\ & \mu_i^{Obs} \texttt{ covers } P_i^+ \texttt{ and zero} - \texttt{covers } P_i^-\} \end{aligned} \tag{5.1}$$

Local diagnoses are obtained by restricting the calculated initial markings on the source places modeling local initial states of the corresponding causal model as well as bordered places used as inputs to such a model from the neighboring ones. In fact, the marking of bordered places will be used later for refining the set of local diagnoses.

$$\Delta_i = \{\Pi_{P_i^s}(\mu) \mid \mu \in \mu_i^{ini}\} \tag{5.2}$$

where $\Pi_{P_i^s}$ denotes the restriction on $P_i^s = \{p \mid {}^\bullet p = \emptyset\}$.

Example 5.2. In order to show how diagnoses are computed locally by an agent A_i, consider the example depicted in Figure 5-1 such that A_1 receives the observation "oil light

is green"and that there is a "delay in the acceleration response"from S_1 and A_2 observes that there is "no hole in the oil sump"and the "indicator of temperature is red"from S_2. One of the possible modeling of the observation of A_1 is to mark place $oil\_light(red)$ with an inhibitor (white) token, which signifies that $\neg oil\_light(red) = true$, and place $accel\_resp(del)$ with a normal (black) token. Similarly, the observation of A_2 corresponds to mark place $hole\_oil\_sump(yes)$ by an inhibitor token, and place $temp\_ind(red)$ by a normal one. Such markings correspond to consider the set P_i^- empty and $P_i^+ = Obs_i$ for both local diagnostic problems. It is to be noted that *i*) in the observation of S_1, the value of $ex\_smoke(black)$ is not specified, which signifies that we have an incomplete knowledge about the behavior of S_1; and *ii*) different classifications of the observed findings lead to different diagnostic problem definitions as it has been discussed in [18], where the authors suggest that for the same observation, we may have a spectrum of definitions varying from a pure consistency-based to a pure abductive diagnosis. Figure 5-2 shows the backward reachability graph obtained by A_1 during local explanation of its received observation. Notation $p[b]$ means that place p is marked with a black token and $p[w]$ that p is marked with a white token; arcs are labeled with the set of transitions that are fired (in backward fashion); the negation symbol is used for transitions that are fired with inhibitor tokens and underlined transitions represent forced ones.

The initial marking μ_1^{ini} obtained as a result by A_1 is $<$ oil_sump_state $(worn)[w]$, $pist\_ring(worn)[w]$, $pist\_state(worn)[w]$, $\boldsymbol{x}[w]$, $\boldsymbol{z}[b]$ $>$. By projecting μ_1^{ini} on the places modeling initial states of the causal model of S_1, we obtain:

$$\Delta_1 =< oil\_sump\_state(worn)[w], pist\_ring(worn)\ [w], pist\_state(worn)[w] >$$

which means that the observed manifestations are explained locally by the absence of local failures. Moreover, μ_1^{ini} specifies that places x and z must be marked by white and black tokens respectively, which means that such tokens have been entered in BPN_1 from BPN_2. We will show later how this information will be used to refine the set of diagnoses.

Similarly, agent A_2 constructs the backward reachability graph of BPN_2 from the marking $\mu_2^{Obs_2}$. The initial markings obtained as results are the following three markings:

$$\mu_2^{ini_1} =< road\_cond(poor)[w], ground\_clear(low)[w], spark\_plug\_mileage(high)[b], \boldsymbol{y}[b] >;$$

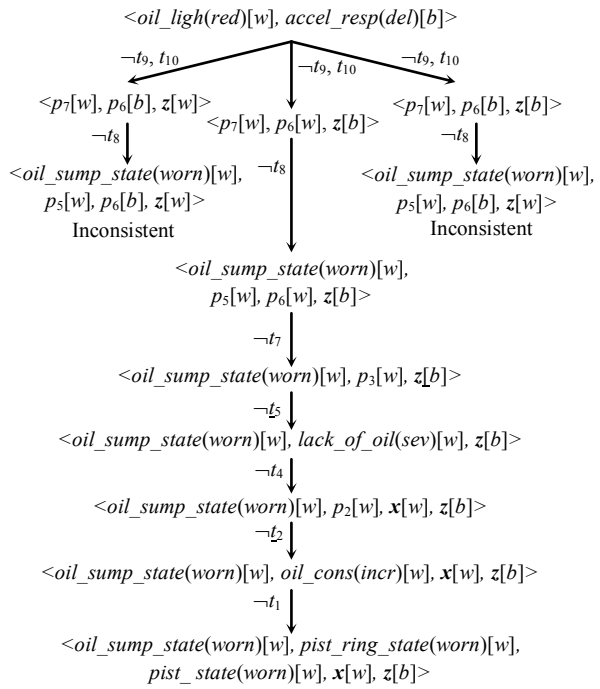

Figure 5-2: The BW graph of A_1.

$$\mu_2^{ini_2} =< road\_cond(poor)[w], ground\_clear(low)[w], spark\_plug\_mileage(high)[w], \boldsymbol{y}[b] >;$$
$$\mu_2^{ini_3} =< road\_cond(poor)[w], ground\_clear(low)[w], spark\_plug\_mileage(high)[b], \boldsymbol{y}[w] > .$$

Thus, agent A_2 obtains three local diagnoses, the first two indicate that BPN_2 must receive a normal token in place $\boldsymbol{y}$ from S_1; while the later one necessitates an inhibitor token in $\boldsymbol{y}$.

Once local diagnoses are obtained, agents begin to communicate among them according to the markings of common places. In particular, each agent asks its neighbors for the required marking of its input places. This is accomplished by sending for each local diagnosis a message Msg containing the marking of its input places.

$$Msg_{i\to j} = \mu_{i\to j}^{Msg} = \{\Pi_{P_i^{In}\cap P_{ij}}(\mu) \mid \mu \in \mu_i^{ini}\} \tag{5.3}$$

Each agent, after receiving such a message, will guarantee that at least one of its local diagnoses is in conformance with the received marking. The following section describes how agents act in this case.

5.3.2 A protocol for the distributed BW-Analysis

In order to ensure that the obtained local diagnoses recover completely global ones that would be computed by a centralized agent that knows the system's global model and receives all manifestation signalizations, each agent must guarantee that the required tokens in its input places are supported by tokens produced by neighboring agents. Otherwise, there is an inconsistency between diagnoses of the corresponding agents. To do so, each agent A_i tries to predict for each obtained μ_i^{ini} (i.e. for each local diagnosis) the marking of its output places in order to compare them with any received message. Local diagnoses for which such markings are not consistent with the neighborhood are simply discarded, because they are not supported by any of the local diagnoses of the other agents and consequently they do not belong to any of the global diagnoses.

For predicting the marking of output places from an initial marking μ_i^{ini}, agent A_i constructs the corresponding marking graph. Unlike the step of local calculations, such a graph is constructed in a forward fashion, because we need to know the marking of output places from a known initial marking [60]. Consequently, the graph construction will terminate when the status of output places is known, since we do not need to calculate all reachable markings. This can be formalized as follows:

$$\mu_i^{Out} = \{\Pi_{P_i^{Out}}(\mu) \mid \mu \in [\mu_i^{ini}\rangle \wedge p \in P_i^{Out} \Rightarrow \mu(p) \neq 0\} \tag{5.4}$$

As a final result, agent A_i compares each received message $Msg_{j\to i}$ from A_j with the obtained markings of output places μ_i^{Out} from each μ_i^{ini}. If for the marking encoded in $Msg_{j\to i}$ there exists at least one marking μ_i^{Out} such that $\forall p \in P_{ij}, \mu_{j\to i}^{Msg}(p) = \mu_i^{Out}(p)$ where P_{ij} denotes common places between BPN models of A_i and A_j, then A_i responds to the message sent by A_j by a positive response, which means that the A_j's local diagnosis from which $Msg_{j\to i}$ has been generated is supported by diagnoses of A_i. Conversely, if from all μ_i^{ini} there does not exist any marking μ_i^{Out} that supports $Msg_{j\to i}$, then A_i will respond to A_j by a negative response. Consequently, when agent A_j receives a negative response to the message $Msg_{j\to i}$ from an adjacent agent, it will discard its local diagnosis Δ_j from which $Msg_{j\to i}$ has been generated, since it do not conform to diagnoses of the neighborhood

even if it explains locally the observed misbehavior. Moreover, it may be that the discarded diagnosis has been used to validate consistency between diagnoses of A_j and those of another adjacent agent A_k (i.e. $k \neq i$). As a result, some of A_k's diagnoses should be eliminated since they become inconsistent with diagnoses of A_j; and thus the communication between agents will be initiated again. Accordingly, the consistency checking will terminate after some communication rounds when a stability condition in terms of local diagnoses of all agents is achieved.

The recovering of global results that would be derived by a centralized agent by those derived by local ones can now be captured by the following proposition:

Proposition 1 *Let Δ be the set of global diagnoses corresponding to an overall diagnostic problem $DP = (N, \langle P^+, P^- \rangle)$ that would be derived by a centralized agent; and let Δ_i be the set of local diagnoses corresponding to a local problem $DP_i = (N_i, \langle P_i^{In}, P_i^{Out} \rangle, \langle P_i^+, P_i^- \rangle)$ that are derived by a local agent A_i with $DP = \bigcup_{i=1}^{n} DP_i$; then: $\Delta_i = \Pi_{N_i}(\Delta), \forall i = 1..n$.*

Proof. The proof follows from the fact that both global and (respec. local) diagnoses are given in terms of the status of global (respec. local) source places by applying the same backward firing rules from a global (respec. local) final marking. ■

It is to be noted that each round of information exchange between neighboring agents attempts to ensure what [82] called the local consistency among local solutions in the context of DES diagnosis. The global consistency property of [82] is achieved when the protocol terminates.

Example 5.3. Consider our example, discussed previously, to show how agents A_1 and A_2 update their diagnoses. After local calculations, each agent sends to the other a message encoding the required marking of places used as its inputs. Thus, agent A_1 sends to A_2 the message ($\boldsymbol{x} = w$, $\boldsymbol{z} = b$). Similarly, A_2 sends for each obtained μ_2^{ini} the required marking of $\boldsymbol{y}$ to A_1. Then, each agent tries to construct its reachability graph in a forward fashion to predict the marking of its output places given the initial marking. Thus, A_1 will construct its reachability graph from $\mu_1^{ini} = < oil\_sump\_state(worn)[w], pist\_ring(worn)[w], pist\_state(worn)[w], \boldsymbol{x}[w], \boldsymbol{z}[b] >$. As a result, the marking of $\boldsymbol{y}$ is obtained to be a white

token. Consequently, A_1 responds to the received requests from A_2 by sending a positive response to the message corresponding to $\mu_2^{ini_3}$, and a negative one for the other two messages. Accordingly, A_2 retains $\mu_2^{ini_3}$ as the only legal diagnosis, since $\mu_2^{ini_1}$ and $\mu_2^{ini_2}$ require that $\boldsymbol{y}$ will be marked by a normal token which is inconsistent with knowledge of A_1. Independently, A_2 will construct for each obtained μ_2^{ini} the corresponding forward reachability graph in order to predict the marking of its output places. According to the three local initial markings given in the last section $\mu_2^{ini_1}$, $\mu_2^{ini_2}$, and $\mu_2^{ini_3}$, the marking of $\boldsymbol{x}$ will be an inhibitor token and that of $\boldsymbol{z}$ is a normal one. These values correspond to those requested by A_1. In other words, the local diagnosis of A_1 is consistent with the results obtained by A_2. In contrast, not all diagnoses of A_2 are consistent with results of A_1.

As a final result, the observed misbehavior of the whole system is explained by the following diagnoses: Δ_1 concerns the subsystem S_1 and Δ_2 concerns S_2.

5.4 Conclusion

In this chapter, a distributed causal model-based diagnosis approach was defined. The approach uses BPNs with common bordered places to describe the causal faulty behavior of a distributed system. It is a state based approach in the sense that the observation is modeled as a set of manifestations to be generated when the system to be diagnosed reaches a particular state. The diagnosis reasoning mechanism is implemented locally as a backward reachability problem within each diagnostic agent. The global consistency between diagnoses of the different agents is achieved through exchanging a limited information about the marking of common places between neighboring BPN models. This requires to construct in a forward fashion the reachability graph from each local diagnosis to seek if the received marking is reachable or not.

Chapter 6

A distributed P-invariant analysis

6.1 Introduction

Even if the modular analysis of Petri net models has been shown powerful to attack the storage and the time complexity, the BW-Analysis and its distributed version described previously suffer from the so-called state space explosion problem even for small net models. This is due to the utilization of reachability graphs as a reasoning scheme especially in the consistency checking phase for the distributed version where several graphs may be constructed by each agent.

This chapter focuses on structural analysis of net models instead of reachability graphs to implement diagnostic reasoning. More particularly, we attempt to relate the set of diagnoses to that of P-invariants minimal supports of BPN models. Such supports are to be generated in an off-line manner and will be used to the on-line diagnosis of the system to be diagnosed

The chapter is organized as follows: we start in section 6.2 by showing how to relate the local diagnoses derived by each agent to the set of minimal supports of P-invariants of the corresponding BPN model. Such a relation exploits an idea known in answer extraction in Petri net models of logic programs. Then in section 6.3, we show how agents cooperate among them to ensure the global consistency between their computed local diagnoses. Finally, section 6.4 concludes the chapter.

6.2 Local diagnosis by analyzing P-invariants

Another alternative to obtain the set μ_i^{ini} is to exploit structural properties of net models. As we have outlined, the aim of this chapter is to concentrate on invariant analysis to realize diagnostic inference procedures rather than reachability graphs. In particular, we will concentrate on how to generate initial markings satisfying the conditions of Eq.(5.1) from a set of P-invariant supports.

For each $\mu \in \Delta_i$, let $diag_i = \{p \mid \mu(p) \neq 0\}$ be the marked places in each local diagnosis; then $Diag_i = \bigcup_{\mu \in \Delta_i} \{diag_i\}$ is a compact representation of Δ_i in which each diagnosis $diag_i$ is viewed as a set of source places rather than a marking.

By definition, P-invariants of a net $N = \langle P, T, F \rangle$ correspond to T-invariants of its dual net $N_D = \langle T, P, F \rangle$. The following lemma has been proved in [53, 65]:

Lemma 1 *Let $N = \langle P, T, F \rangle$ be a Petri net such that $\forall t \in T, |t^\bullet| \leq 1$ and $t \in T$ be a sink transition; there exists a T-invariant X of N such that $X(t) \neq 0$ iff t is potentially firable from the empty marking.*

This means that in N there are source transitions firing from the empty marking (i.e. $\forall p \in P : \mu(p) = 0$), eventually leading to the firing of t. Consider now the dual net of N, $\forall t \in T |t^\bullet| \leq 1$ becomes $\forall p \in P |p^\bullet| \leq 1$, which is guaranteed by axiom 1 of Definition 15 given in chapter 3, and the previous lemma can be translated as follows:

Proposition 1 *Let $N = \langle P, T, F \rangle$ be a Petri net such that $\forall p \in P, |p^\bullet| \leq 1$, and $p \in P$ be a sink place; there exists a P-invariant Y of N such that $Y(p) \neq 0$ iff p can potentially be marked by firing a sequence of transitions from an initial marking μ in which $\mu(p) \neq 0 \Rightarrow {}^\bullet p = \emptyset$.*

Proof. This is a consequence of Lemma 1, and the fact that P-invariants of a net are T-invariants of its dual net. ■

This proposition means that the supports of the P-invariants of a net N modeling a causal behavior of a system characterize the diagnostic solutions that explain a set of manifestations. In fact, [68] proposes an algorithm based on analyzing such supports for the centralized causal

model-based diagnosis on BPNs. Before applying the algorithm, [68] requires to transform, via an $\wedge$-fusion operation, the BPN model to another equivalent net model in which places that are "And-ed"in the original BPN are collapsed into a single place representing their conjunction. More formally:

$\forall t \in T_N$: if ${}^\bullet t = \{p_1, ..., p_k\}(k > 1)$ then substitute in P the set $\{p_1, ..., p_k\}$ with the place $p_{1,k}$ such that ${}^\bullet p_{1,k} = \bigcup_{i=1}^{k} {}^\bullet p_i$ and $p_{1,k}^\bullet = \{t\}$. It is to be noted that such a transformation is needed only for getting a right interpretation of P-invariants; and that even if the resulting net is no longer a BPN, it encodes the same kind of knowledge of the original BPN.

The algorithm can now be sketched as follows: after having calculating the minimal supports of P-invariants of the $\wedge$-fusion transform of the net model, those leading to mark places in P^- (i.e. $\{\sigma \mid p \in P^- \wedge p \in \sigma\}$) are eliminated by taking into account the fact that if τ, τ' are two sets of source places such that $\tau \subseteq \tau'$, if the marking of τ leads to mark $p \in P^-$, then the marking of τ' leads also to mark p. Then, the algorithm considers the coverability of P^+; for each $p \in P^+$, it builds from remaining supports the list of source places (i.e. places denoting initial states of the causal model) supporting p (i.e. contained in a P-invariant support containing p). Final diagnoses $Diag_i$ are obtained by combining such lists.

Example 6.1. Let us change our example depicted in Figure 5-1 so that transition t_1 becomes an Or-transition. Remember that the observation of A_1 corresponds to $oil\_light(green)$ and $accel\_resp(del)$ (i.e. "oil light is green" and there is a "delay in the acceleration response") and that of A_2 corresponds to $hole\_oil\_sump(no)$ and $temp\_ind(red)$ (i.e. there is "no hole in the oil sump" and the "indicator of temperature is red"). Let us suppose that all abnormal observations have to be covered, then:

$P_1^+ = \{accel\_resp(del)\}, P_1^- = \{oil\_light(red)\}$;
$P_2^+ = \{temp\_ind(red)\}, P_2^- = \{hole\_oil\_sump(yes)\}$.

Notice that: *i*) $oil\_light(green)$ is not represented in the model because it is not part of the faulty behavior of S_1. However, this leads to put the place $oil\_light(red)$ in the set P_1^- meaning that the instance red of the finding oil_light contradicts the observed value; and *ii*) the $\wedge$-fusion transformation results in replacing places $oil\_sump\_state(worn)$ and p_5 by

place $p_{oil\_sump\_state(worn),p_5}$ in BPN_1.

The set of minimal supports of P-invariants of S_1 that are computed by A_1 are the following:

$\sigma_1 = \{pist\_ring\_state(worn),\ oil\_cons(incr),\ p_1,\ ex\_smoke(black)\}$;

$\sigma_2 = \{pist\_state(worn),\ oil\_cons(incr),\ p_1,\ ex\_smoke(black)\}$;

$\sigma_3 = \{pist\_ring\_state(worn),\ oil\_cons(incr),\ p_2,\ lack\_of\_oil(sev),\ p_4,\ \boldsymbol{y}\}$;

$\sigma_4 = \{pist\_state(worn),\ oil\_cons(incr),\ p_2,\ lack\_of\_oil(sev),\ p_4,\ \boldsymbol{y}\}$;

$\sigma_5 = \{pist\_ring\_state(worn),\ oil\_cons(incr),\ p_2,\ lack\_of\_oil(sev),\ p_3,\ p_6,\ accel\_resp(del)\}$;

$\sigma_6 = \{pist\_state(worn),\ oil\_cons(incr),\ p_2,\ lack\_of\_oil(sev),\ p_3,\ p_6,\ accel\_resp(del)\}$;

$\sigma_7 = \{pist\_ring\_state(worn), oil\_cons(incr), p_2, lack\_of\_oil(sev), p_3, p_7, p_{oil\_sump\_state(worn),p_5},$
$oil\_light(red)\}$;

$\sigma_8 = \{pist\_state(worn),\ oil\_cons(incr),\ p_2,\ lack\_of\_oil(sev),\ p_3,\ p_{oil\_sump\_state(worn),p_5},\ p_7,$
$oil\_light(red)\}$;

$\sigma_9 = \{\boldsymbol{x},\ lack\_of\_oil(sev),\ p_4,\ \boldsymbol{y}\}$;

$\sigma_{10} = \{\boldsymbol{x}, lack\_of\_oil(sev),\ p_3,\ p_6,\ accel\_resp(del)\}$;

$\sigma_{11} = \{\boldsymbol{x}, lack\_of\_oil(sev),\ p_3,\ p_{oil\_sump\_state(worn),p_5},\ p_7,\ oil\_light(red)\}$;

$\sigma_{12} = \{\boldsymbol{z},\ accel\_resp(del)\}$.

Since place $oil\_light(red) \in P_1^-$, any support predicting the marking of such a place will be eliminated; hence, σ_7, σ_8 and σ_{11} will be discarded because they contain $oil\_light(red)$. Moreover, any support that contains one of the places $pist\_ring\_state(worn)$, $pist\_state(worn)$ and $\boldsymbol{x}$ will be eliminated, since the marking of one of these places conducts to mark $oil\_light(red)$ according to the discarded supports. Thus, the only support that survives is σ_{12} which explain locally the marking of $accel\_resp(del)$ by the arrival of a token in place $\boldsymbol{z}$. In other words, the observed local findings are explained by an outside failure propagated to S_1 through $\boldsymbol{z}$.

Similarly, for A_2 seven minimal supports of P-invariants of S_2 are generated:

$\varsigma_1 = \{road\_cond(poor),\ p_8,\ \boldsymbol{x}\}$;

$\varsigma_2 = \{road\_cond(poor),\ p_8,\ p_9,\ hole\_oil\_sump(yes)\}$;

$\varsigma_3 = \{ground\_clear(low),\ p_{10},\ hole\_oil\_sump(yes)\}$;

$\varsigma_4 = \{spark\_plug\_mileage(high), spark\_plugs(used\_up), p_{11}, spark\_ign(irr), temp\_ind(red)\}$;

$\varsigma_5 = \{spark\_plug\_mileage(high),\ spark\_plugs(used\_up),\ p_{11},\ \boldsymbol{z}\}$;

$\varsigma_6 = \{\boldsymbol{y},\ p_{11},\ spark\_ign(irr),\ temp\_ind(red)\}$;

$\varsigma_7 = \{\boldsymbol{y},\ p_{11},\ \boldsymbol{z}\}$.

Supports ς_2, ς_3 are discarded because they contain $hole\_oil\_sump(yes)$ which belongs to P_2^-. Since ς_1 contains $road\_cond(poor)$ which is contained with $hole\_oil\_sump(yes)$ in the same support, it will be also eliminated. The remaining supports will be used by A_2 to generate diagnoses that explain locally the marking of $temp\_ind(red)$.

As a result, the observed symptom is explained by one of the following two local diagnoses:

$diag_2^1 = \{spark\_plug\_mileage(high)\}$;

$diag_2^2 = \{\boldsymbol{y}\}$.

The first one means that $temp\_ind(red)$ is caused by a local failure; while the latter signifies that there is a failure in the neighborhood affecting the behavior of S_2 through $\boldsymbol{y}$.

Thus, the general diagnostic algorithm based on the P-invariant analysis used by each agent A_i to compute local solutions can be sketched informally as follows:

Algorithm 1: *A_i's Local Computation*

Input: a local diagnostic problem in terms of Petri nets

$DP_i = (N_i, \langle P_i^{In}, P_i^{Out}\rangle, \langle P_i^+, P_i^-\rangle)$;

Output: a set of local diagnoses in terms of minimal supports;

begin

compute the minimal supports of the P-invariants for N_i;

let L be the list of such minimal supports;

for each $p \in P_i^-$ **do**

for each support $\sigma \mid p \in \sigma$ **do** // for each σ such that $p \in \sigma$

$\tau \leftarrow \{p' \in \sigma \mid {}^\bullet p' = \emptyset\}$;

delete from L all supports where τ occurs;

end for

end for

end.

It is to be noted that in the above algorithm, N_i is considered as the $\wedge$-fusion transform of the original BPN model, and that the local diagnoses can be obtained by combining source places belonging to the remaining supports. We choose to not combine such supports during this first step of pruning because they are needed by agents during the consistency checking step.

6.3 Protocol for distributed P-invariant analysis

To ensure global consistency between local diagnoses of the different agents, each one asks their neighbors for the required set of its input places that need to be marked (i.e. that necessitate to receive tokens from neighboring net models). According to Eq.(5.1), such places can be obtained by choosing the marked places from the results of projecting the set μ_i^{ini} on P_i^{In} (i.e. $\{p \mid \forall p \in P_i : \mu(p) \neq 0 \wedge \mu \in \Pi_{P_i^{In}}(\mu_i^{ini})\}$) which is equivalent to choose the marked input places from Δ_i. Thus, agent A_i will send to each of its neighbors a message indicating what input places are used to explain the local observation for each of its obtained local diagnoses Δ_i.

$$Msg_{i \to j} = \{p \mid p \in P_i^{In} \cap P_{ij} \wedge \mu(p) \neq 0 \wedge \mu \in \Delta_i\} \tag{6.1}$$

Equation 6.1 is given in terms of the set of reachable markings. Because we make use of the set of minimal supports of P-invariants as a basis on which diagnosis is accomplished and not directly on the net models and on the associated reachability graphs, we wish to exploit such supports in order to check the required consistency; and hence avoiding the combinatorial explosion of state space characterizing reachability graphs. Thus, equation 6.1 can be transformed as:

$$Msg_{i \to j} = \{p \mid p \in P_i^{In} \cap P_{ij} \cap \sigma \wedge \sigma \in L\} \tag{6.2}$$

where L is the set of N_i'minimal supports pruned by Algorithm 1.

In this spirit, when agent A_j receives a message $Msg_{i \to j}$, it will examine its remaining set of supports to check if places contained in the received message belong at least to one

of such supports. If so, it will respond, as in the distributed BW-Analysis, by a positive response indicating that diagnoses of A_j are consistent with the diagnosis of A_i from which $Msg_{i\to j}$ has been generated. Otherwise, A_j's local diagnoses do not support the diagnosis Δ_i of A_i; and hence, a negative response should be sent. Reception of responses by agents is handled in the same manner as in the distributed BW-Analysis.

Hence, Proposition 1 of chapter 5 can be rewritten as follows:

Proposition 2 *Let $Diag$ be the set of global diagnoses corresponding to an overall diagnostic problem $DP = (N, \langle P^+, P^- \rangle)$ that would be derived by a centralized agent; and let $Diag_i$ be the set of local diagnoses corresponding to a local problem $DP_i = (N_i, \langle P_i^{In}, P_i^{Out} \rangle, \langle P_i^+, P_i^- \rangle)$ that are derived by a local agent A_i with $DP = \bigcup_{i=1}^{n} DP_i$; then: $Diag_i = \Pi_{N_i}(Diag), \forall i = 1..n$.*

Proof. Since $Diag$ and $Diag_i$ are simplified representations of Δ and Δ_i respectively, the proof becomes obvious from Proposition 1 of page 70. ■

We can now extend our previous algorithm of local computation to account for message exchange between the different agents. For simplicity purposes, the algorithm will be presented in three parts: the first part (Algorithm 2) extends Algorithm 1 to generate messages that will be sent to the neighborhood and to eliminate local solutions for which it receives a negative response. The second part (Algorithm 3) treats the case of a message reception by a neighboring agent A_j. Finally, Algorithm 4 generates local diagnoses that are globally consistent and may be viewed as the last task of each diagnostic agent to be executed after the completion of the communication protocol.

Algorithm 2: *A_i's Local computation with communication*

Input: the list L of minimal supports pruned by Algorithm 1;

Output: a list of minimal supports that are consistent with those of the neighborhood;

begin

$Msg\_list \leftarrow \emptyset$;

for each $\sigma \mid p \in \sigma \cap P_i^{In}$ **do**

if $(Msg_{i\to j}, \sigma') \notin Msg\_list \mid Msg_{i\to j} = p$ **then**

$Msg_{i\to j} \leftarrow \{p\}$ where j is such that $p \in P_j^{Out}$;

Send $Msg_{i\to j}$ to A_j;

$Msg\_list \leftarrow Msg\_list \cup \{(Msg_{i\to j}, \sigma)\}$;

end if

end for

while $Msg\_list \neq \emptyset$ **do**

receive(rep);

let rep be the response corresponding to $(Msg_{i\to j}, \sigma)$;

if $rep = negative$ **then**

delete from L all supports where $Msg_{i\to j}$ occurs;

end if

$Msg\_list \leftarrow Msg\_list \backslash \{(Msg_{i\to j}, \sigma)\}$;

end while

end.

Algorithm 3: *Treatment of a received message*

Input: a received message $Msg_{i\to j}$;

Output: a positive or a negative response;

begin

if $\exists \sigma \in L \mid Msg_{i\to j} \in \sigma$ **then**

reply to $Msg_{i\to j}$ with a positive response;

else

reply to $Msg_{i\to j}$ with a negative response;

end if

end.

Algorithm 4: *Diagnoses generation*

Input: the list L of minimal supports pruned by Algorithm 1;

Output: local diagnoses that are globally consistent;

begin

$X \leftarrow \emptyset$;

for each $p \in P^+$ **do**

if $\nexists \sigma \in L \mid p \in \sigma$ **then** no solution;

$H' \leftarrow \emptyset$;

for each $\sigma \in L \mid p \in \sigma$ **do**

$H' \leftarrow H' \cup \{p' \mid p' \in \sigma \wedge {}^\bullet p' = \emptyset\}$;

end for

$X \leftarrow X \cup \{H'\}$;

end for

combine elements in X to produce $Diag_i$;

end.

The proof that the above algorithms are sound follows from Proposition 2. It means that the set L of minimal supports of the P-invariants of a global BPN is recovered by all agents through distributed calculations. In effect, L corresponds to the composed union over common bordered places of the sets L_i of all local BPNs. In particular, let $p_{m(v)} \in P^+$ be a sink place corresponding to a manifestation instance $m(v)$ which is observed by the centralized agent, then $p_{m(v)}$ is certainly present in at least one of the supports of L. If $p_{m(v)} \in P_i^+$ (i.e. if $m(v)$ is within DP_i), then $p_{m(v)}$ is also present in at least one support σ of L_i. In this case, $m(v)$ is explained either by a local failure in the associated subsystem S_i or by a failure in the neighborhood (i.e. $p_{m(v)}$ is present with some common places in σ). The proof that the above algorithms terminate (i.e. actually Algorithm 2) after finitely many communication rounds relies on the assumption that the BPN models are safe and their composed global one is acyclic (Assumption 1).

Example 6.2. In order to show how agents update their diagnoses, let us return again to our example. After local computations, each agent sends to the other a message that contains the required places used as its inputs for each of its local diagnoses. Thus, agent A_1 sends to A_2 the message $Msg_{1\to 2} = \{\boldsymbol{z}\}$ (Notice that $\boldsymbol{z}$ has been identified as the only local diagnosis for the observation of A_1). Similarly, A_2 sends $Msg_{2\to 1} = \{\boldsymbol{y}\}$ to agent A_1. Then, each agent tries to test if the received common place belongs to its remaining set of supports. Thus, A_1 concludes that there is an inconsistency between its diagnosis and

knowledge of A_2; because its list of supports contains only σ_{12} and $\boldsymbol{y} \notin \sigma_{12}$. Consequently, it responds by a negative response; and accordingly, A_2 will discard ς_6 and ς_7. Independently, when A_2 receives $Msg_{1\rightarrow 2} = \{\boldsymbol{z}\}$, it concludes immediately that $\boldsymbol{z}$ belongs to its supports ς_5 and ς_7 (i.e. $\boldsymbol{z}$ belongs to the remaining supports even after eliminating ς_7), and it responds with a positive response. Since this first round of communication has resulted in a modification of A_2's diagnoses; agents will restart the communication again. During the second communication round, local diagnoses remain unchanged; and the process terminates with σ_{12} as the only support for A_1 and $\{\varsigma_4, \varsigma_5\}$ for A_2. This means that $Diag_1 = \{\boldsymbol{z}\}$ (i.e. for A_1, the local observed symptom $accel\_resp(del)$ is caused by a failure in the neighborhood which is propagated to S_1 through $\boldsymbol{z}$), and $Diag_2 = \{spark\_plug\_mileage(high)\}$ (i.e. the $temp\_ind(red)$ observed by A_2 is explained by $spark\_plug\_mileage(high)$).

6.4 Conclusion

Besides local computations, the consistency checking is also accomplished through exploiting structural properties of the BPN models. In effect, when an agent receives a message, it seeks if the content of such a message belongs to its remaining supports. Thus, instead of constructing a set of reachability graphs for each received message, the receiver agent will make a simple belonging test to respond to the sender one.

Chapter 7

Relationships among manifestations

7.1 Introduction

A close inspection of the analysis techniques presented in previous chapters (chapters 5 and 6) concludes that they make abstraction from relationships that may exist among manifestations during their signalization, since all the observed findings are gathered at a single time point which is the starting moment of the diagnostic process. This is due to the fact that any instance of a finding (i.e. a manifestation state) in the causal model is represented by a sink place in the corresponding net model; that is, BPNs have been used without taking into account the precedence of occurrences between the generated symptoms. The aim of this chapter is to extend the work presented in these chapters to handle such relationships in the sense that the made observation is viewed as a (partially ordered) observable sequence (i.e. Definition 5 given in page 38). This will necessitates, on one hand the representation of these relations in the BPN model; and on the other hand the definition of other firing rules based on BPNs for the BW-Analysis and consequently the adaptation of the invariant analysis of [68] to account for such relations. Moreover, real world applications of diagnosis, such as fault management in telecommunication networks, are characterized by the fact that some of the generated manifestations may be lost or suppressed. Consequently, the diagnosis process may not be able to explain the observed misbehavior. In terms of a BPN model, the BW-Analysis may ends within an inconsistent marking and that of invariants within an empty set of supports. In order to handle such phenomenon, the retained solution consists to

slightly change the set of observed manifestation instances so that the given problem admits a particular interpretation model.

We start this chapter by showing in section 7.2 how to introduce in a BPN model precedence relationships that may exist between manifestation instances. Then in section 7.3, we propose a novel set of backward firing rules to be considered in conjunction with those of figure 3-2 for diagnosis based on reachability graphs. Section 7.4 considers the invariant-based technique. In particular, we will motivate the adaptation of our modelization of such relationships in order to apply the P-invariant analysis technique. The problem of manifestation losses and how to handle it is signaled briefly in section 7.5. Finally, section 7.6 concludes the chapter.

7.2 Relationships model

From section 3.4.3, one can conclude that in BPN models relationships among manifestation states are inexistent. In order to model this kind of relations, we proceed as in [4] when we have introduced in PN models transitions among places representing manifestation instances to model occurrence orders between symptoms. More formally, if from a given state instance s_1, the manifestation instance m_j will be generated after generating m_i; this may be modeled by adding a transition from the place corresponding to m_i to that corresponding to m_j. Moreover, the added transition will be an Or-transition, if m_j can be caused by another state different than s_1; otherwise, it will be an and-transition (i.e. a linear transition). Note that if m_i and m_j are to be generated without any order's constraint (i.e. independently) when the system reaches the state s_1, then no transition is added since the semantics of fork transitions in BPNs allow to model such independency. Figure 7-1 summarizes these graphically.

7.3 Extending the BW-Analysis technique

The backward firing rules defined in [2] (see figure 3-2) make abstraction from dependency relationships that may exist among fault indications; and consequently the construction of

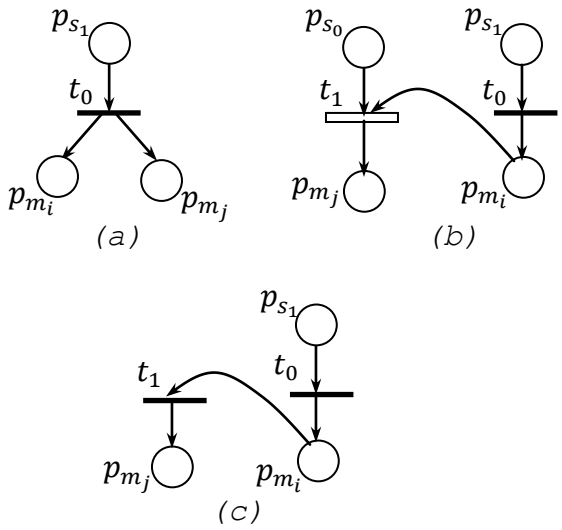

Figure 7-1: Relationships that may exist between manifestation instances.

the markings graph requires that the *b-w* marking from which it is started is such that the marked places are only sink ones (places representing manifestation states of the causal model). Because we allow manifestations to be modeled by any place (either a sink or a not sink place) for introducing in the net model relationships among them, it is necessary to define other firing rules to account for such relations. In effect, when a set of symptoms have to be generated in a given order from a particular state instance, the diagnosis process should not identify as diagnostic solutions other causes that are different from the primary source of such state instance.

The possible relationships of occurrences among manifestations are of two types: dependency and independency relations. For the latter case, the firing rules of figure 3-2 remain valid. In order to handle dependency relationships between manifestations, we need to look in the behavioral model if such dependencies exist. If so, the above firing rules cannot explain the observed misbehavior. In effect, according to the modeling manner presented previously, if the symptom m_j is observed present (respectively absent) after observing m_i, then the two places corresponding to such manifestations are marked with normal (respectively inhibitor) tokens. Thus, the backward construction of the markings graph will remove token from the place corresponding to m_j and puts it into the place corresponding to m_i. Nevertheless, because the BPN model is safe, any place will contain at most one token, and so the removed token is consumed. Thus, the following firing rules given graphically in figure 7-2 are used

to construct the markings graph.

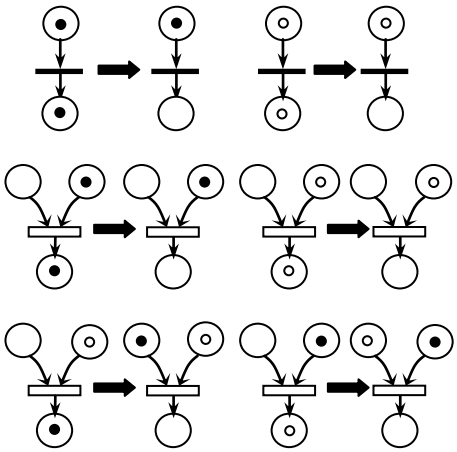

Figure 7-2: Backward firing rules taking into account relationships among manifestations.

The meaning of the first two rules is that if a symptom m_j will be generated only when another symptom m_i has been generated, then explaining m_j becomes a question to explain m_i. For the case where m_j may be generated either after generating another symptom m_i or when the system reaches another state (that is, independently from m_i), then if we observe m_j we will verify if m_i has been observed. If so, then explaining m_j is equivalent to explaining m_i. This is what the next two rules of Figure 7-2 denote. The last two rules of figure 7-2 symbolize the fact that when m_j has been observed present (a black token) and that m_i has been found to be absent (a white token) or vice versa, then explaining such observation necessitates to explain each symptom independently. Let us illustrate how these firing rules can be exploited to solve diagnostic problems.

Example 7.1. Consider the net model of figure 7-3 with places p_{m_1} and p_{m_2} are marked with normal tokens and p_{m_3} is marked with an inhibitor token. This corresponds to observing the presence of p_{m_1} and p_{m_2} and the absence of p_{m_3}. Let us suppose that the observation is received in the order: $m_2, m_1, \neg m_3$ (i.e. as a sequence of findings). A backward reachability graph can be obtained by applying the above firing rules as shown in figure 7-4. In this graph, the observed misbehavior is explained by the state instance c_2 and the negation of the state instance c_3. It is to be noted that the instance c_2 suffices to explain both m_1 and m_2 (in the order m_2 followed by m_1).

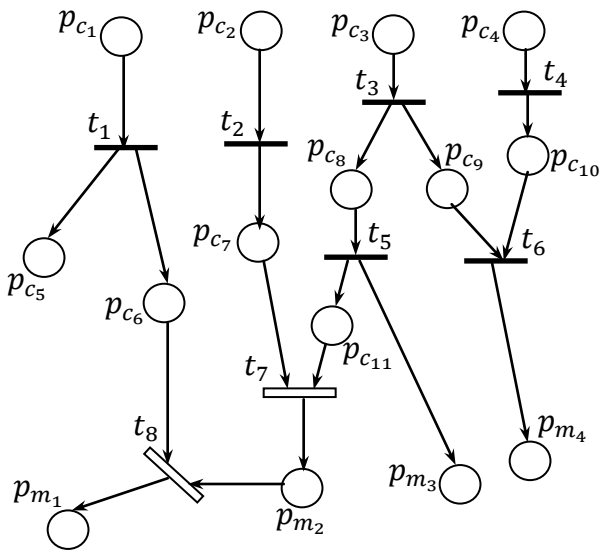

Figure 7-3: Example of a BPN model with relationships among manifestations.

7.4 Extending the P-invariant technique

According to the analysis technique exposed in chapter 6, the basic idea of causal model-based diagnosis by P-invariant analysis consists to discard minimal supports that contain places belonging to the set P^-. Final diagnoses are obtained by combining the lists of source places from the remaining supports. Thus, in order to address the above relationships between manifestation instances, we need to retain the same idea since if two places p_{m_i} and p_{m_j} corresponding respectively to manifestation instances m_i and m_j are within the same support and one of the them is observed present while the other is observed absent, we will discard such a support and consequently the present instance becomes inexplicable.

Hence, the above model of relationships requires to be slightly changed in the sense that any manifestation instance is represented by a sink place. In effect, we need to an artificial manner to model the precedence of occurrences between two manifestations.

7.4.1 Adaptation of the relationships model

Referring to figure 7-1 parts (b) and (c), instead of adding a simple transition from place p_{m_i} to p_{m_j}, we introduce a fork transition t from p_1 such that $|^\bullet t| = p_1 \wedge |t^\bullet| = \{p_{m_i}, \bar{p}_{m_i}\} \wedge |\bar{p}^\bullet_{m_i}| = t_1$. The place $\bar{p}_{m_i}$ is added to keep p_{m_i} a sink one; that is, if t fires, then p_{m_i} and $\bar{p}_{m_i}$ are

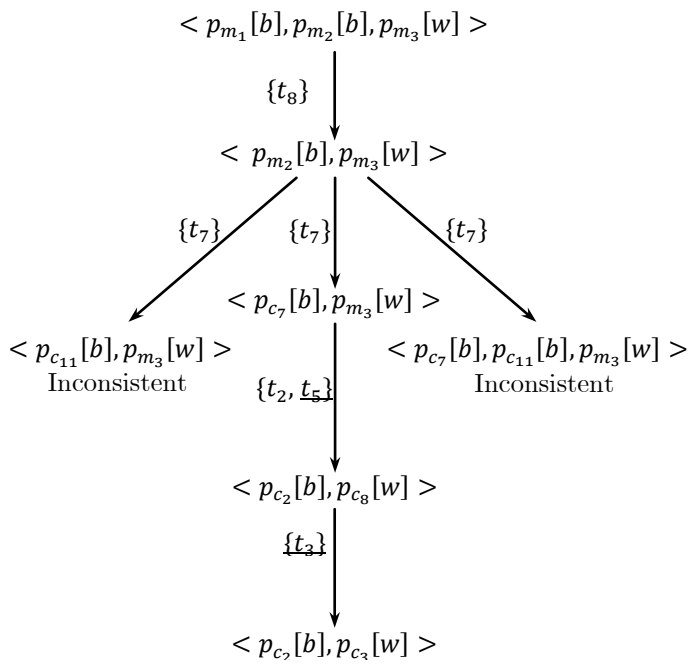

Figure 7-4: The BW graph of Figure 7-3 model.

simultaneously marked before consuming the token of $\bar{p}_{m_i}$ by firing t_1. Figure 7-5 shows such modification graphically.

Example 7.2. As an example, figure 7-6 takes the previous one with the necessary modification of relationships among p_{m_2} and p_{m_1}. In this example, place p is added as an output place of transition t_7 and an input place of t to keep p_{m_2} a sink place.

7.4.2 Diagnosis by P-invariant analysis

As for the case treated by chapter 6, we need to interpret such relationships by analyzing the set of minimal supports of BPN invariants. In effect, the dependency relationships that may exist between two manifestation instances can be reformulated in terms of dependencies among the causes and effects (i.e. the events and conditions in Petri nets terminology) occurred in the system to be diagnosed. In particular, in the case where the manifestation instances m_i and m_j have to be generated in a given order, for example m_i followed by m_j (denoted $m_i \prec m_j$), the set of internal events and conditions that take place before generating m_i will be included into the set of events and conditions that will take place

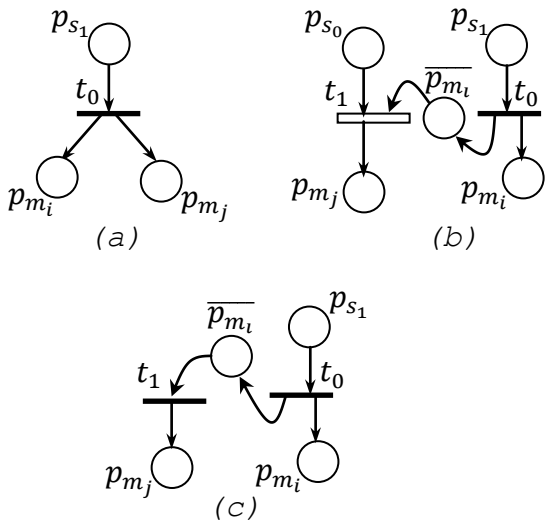

Figure 7-5: A refined model of relationships among manifestations.

before generating m_j. For the case where the two manifestation instances have to generated independently (denoted $m_i \parallel m_j$), the inclusion relation does not hold.

In order to characterize such dependencies, let us introduce the following definition of covering among sets of places.

Definition 1 *A set of places τ_1 covers a set of places τ_2 iff $\tau_1 \neq \tau_2$ and every place in τ_2 occurs in τ_1.*

By considering the minimal supports of P-invariants, we have to characterize the evolutions captured by all the places listed in each support. Therefore, we introduce the following theorem.

Theorem 1 *Let τ_{m_i} be the set of places listed in the same support containing p_{m_i}, and τ_{m_j} be the one listed with p_{m_j}.*

1. *τ_{m_j} covers $\tau_{m_i} \Longrightarrow m_i \prec m_j$.*

2. *any of the two sets τ_{m_i}, τ_{m_j} do not cover the other $\Longrightarrow m_i \parallel m_j$.*

Proof. In fact, the set τ_{m_i} represents the set of internal conditions that will take place before generating the manifestation instance m_i. Since the covering relation consists of an inclusion relation between sets and according to the discussion given in the previous paragraph, the proof of the theorem becomes obvious. ■

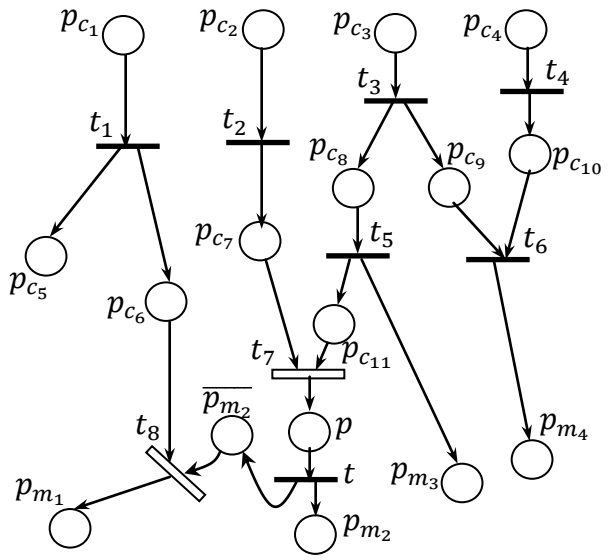

Figure 7-6: A refined example of a BPN model with relationships among manifestations.

Example 7.3. As it is suggested in chapter 6, before computing the P-invariants of a BPN model, we will perform the $\wedge$-fusion transformation of the original net. Consider the BPN of Figure 7-6, in the $\wedge$-fusion transform places p_{c_9} and $p_{c_{10}}$ are collapsed into the place $p_{c_9,c_{10}}$; and the corresponding minimal supports are the followings:

$\sigma_1 = \{p_{c_1}, p_{c_5}\}$;	$\sigma_6 = \{p_{c_3}, p_{c_8}, p_{c_{11}}, p, p_{m_2}\}$;
$\sigma_2 = \{p_{c_1}, p_{c_6}, p_{m_1}\}$;	$\sigma_7 = \{p_{c_3}, p_{c_8}, p_{m_3}\}$;
$\sigma_3 = \{p_{c_2}, p_{c_7}, p, \bar{p}_{m_2}, p_{m_1}\}$;	$\sigma_8 = \{p_{c_3}, p_{c_9,c_{10}}, p_{m_4}\}$;
$\sigma_4 = \{p_{c_2}, p_{c_7}, p, p_{m_2}\}$;	$\sigma_9 = \{p_{c_4}, p_{c_9,c_{10}}, p_{m_4}\}$.
$\sigma_5 = \{p_{c_3}, p_{c_8}, p_{c_{11}}, p, \bar{p}_{m_2}, p_{m_1}\}$;	

Let us consider again the observation sequence $m_2, m_1, \neg m_3$ as in Example 7.1. If we choose to entail all abnormal findings, then $P^+ = \{m_1, m_2\}$ and $P^- = \{m_3\}$. Moreover, since we consider the observation as a sequence of findings, we will consider $P^+ = m_2 m_1$.

By applying Algorithm 1 given in the previous chapter, supports $\sigma_5, \sigma_6, \sigma_7$ and σ_8 are discarded. From remaining supports, we find that σ_2 and σ_3 explain (or entail) the presence of m_1; and σ_4 explains m_2. According to Algorithm 4 of chapter 6 where all observed findings are made in a single time moment, the above observation is explained by combining elements of $H'_1 = \{p_{c_1}, p_{c_2}\}$ and $H'_2 = \{p_{c_2}\}$. In the case where precedence of occurrences among the observed findings is considered, theorem 1 requires to examine not just the sets H' of source

places but all the content of remaining supports where each $p_m \in P^+$ is present. For our example, we will examine supports σ_2, σ_3 and σ_4. From σ_2, the set τ_{m_1} is identified to be $\tau_{m_1} = \{p_{c_1}, p_{c_6}\}$, and that identified from σ_3 is $\tau_{m_1} = \{p_{c_2}, p_{c_7}, p, \bar{p}_{m_2}\}$. Similarly, for p_{m_2}, the set $\tau_{m_2} = \{p_{c_2}, p_{c_7}, p\}$ is identified from σ_4. Comparing τ_{m_2} with the two values of τ_{m_1}, we find that τ_{m_2} is covered by the second value of τ_{m_1} which is identified from σ_3 (i.e. the events and conditions that take place before marking the place p_{m_2} according to σ_4 represent a necessary condition to mark later the place p_{m_1} according to σ_3). As a consequence, σ_2 will be also discarded because the two supports σ_3 and σ_4 explain the presence of m_2 followed by m_2. As a final result, the observed manifestations are caused by the instance c_2.

7.5 Inconsistent markings

Up to here, the proposed analysis techniques both with or without relationships among manifestation instances consider that all the signaled instances of manifestation findings are supposed correctly received. Nevertheless, in real world applications of diagnosis such as fault management in telecommunication networks, the made observation may be altered. That is, some of the generated manifestations may be lost or suppressed. For example, in case of saturation of the telecommunication network (in fact of the management network), some full buffers may provoke losses of alarm signalizations. This is explained notably by the fact that the management information have generally a least priority than the data traffic. Such losses induces "holes"in the observation at the diagnostic agent.

In terms of the BW Analysis, all the alternatives in the backward construction of the reachability graph may end within inconsistent b-w markings. This means that, given a fork transition, it is impossible to have its output places consistently marked with different types of tokens. In the case of P-invariant analysis technique, all the supports may be eliminated. As a result, the set of diagnoses for the received observation becomes empty and hence the observed misbehavior is not explicable. In effect, the holes in the observation OBS may lead to an inconsistency in the given context[1] (i.e. the given diagnostic problem).

In order to handle such inconsistency, we proceed as in [4] when we have looked to the

[1]In fact, losses of manifestation signalizations may lead to a logical inconsistency in $BM \cup OBS$.

detected inconsistency as a result of manifestation masking. The retained solution consists to restore the consistency to the problem in question through an extension of manifestation instances that are observed to be present. Such extension is based on the identification of instances that are known to be absent and that when are supposed masked will restore the required consistency. This can be done by searching in the net model for manifestation places such that for generating one of them the others will be generated.

Example 7.4. As an example, let us suppose to change the observation given in the above example of figure 7-3 such that m_1, m_2, m_4 are observed present and m_3 is observed absent. Thus, the backward analysis starts with $< m_1[b], m_2[b], m_3[w], m_4[b] >$. As a result, we get $< c_2[b], c_4[b], c_8[w], c_9[b] >$ as the last alternative which is an inconsistent b-w marking because c_8 and c_9 are output places of the same fork transition t_3. By analyzing the net model of figure 7-3, we find that m_3 and m_4 will be marked in all cases with the same type of tokens. That is, we conclude that m_3 has been masked; and consequently the diagnosis to the given problem is represented by the b-w marking $< c_2[b], c_3[b], c_4[b] >$.

7.6 Conclusion

For diagnostic problems where the generated manifestations have to be signaled in a given order, the introduction of such orders in the model is essential in order to correctly diagnose the observed misbehavior. Accordingly, the analysis techniques of BPNs presented in previous chapters need to be extended to handle these relationships among manifestations. Finally, we have informally discussed the phenomenon of manifestation masking which may conduct to an inconsistency in the given diagnostic problem and how to face such a phenomenon.

Chapter 8

Conclusion

The advent of large distributed technical systems like computer and telecommunication networks has been one of the most striking developments of our time. Research in distributed model-based diagnosis as documented in several AI conferences and a series of workshops has tackled the question of how to support such systems by a diagnosis architecture similar to that given in 4-1.

In this thesis, we have introduced a BPN-based framework for the diagnosis of spatially distributed systems. The motivation for such a framework is the unnecessary complexity and communication overhead of centralized solutions. Consider a distributed system with n nodes, e.g. a computer network consisting of n machines. When using a centralized diagnosis system, the size of the system description (i.e. number of ground formulas in logical frameworks) is linear in n. Diagnosis time will usually be worse than linear in n. Also all observations have to be transmitted to the central diagnosis machine, causing a large communication overhead.

Our BPN-based approach views the overall system to be diagnosed as a set of interacting subsystems. Each subsystem is diagnosed by an agent which has detailed knowledge over his subsystem and an abstract view of the neighboring subsystems. Most failures can be diagnosed locally within one subsystem. This decreases diagnosis time dramatically in large systems. In the case of the computer network most machines in a subnet can usually fail without affecting machines in other subnets. Only those computers in other subnets can be affected which have sent messages to the faulty machine. Moreover, the local computa-

tion of diagnoses avoids the communication overhead which would be needed to forward all observations to the central diagnosis engine.

Failures, which affect more than one subsystem are diagnosed by the agents cooperating with each other. The cooperation process is triggered locally by an agent, when it realizes that it can not explain the observations by a failure in his own subsystem. The cooperation process is guided by a small amount of topological information. More particularly, local computations are accomplished by analyzing BPN models in a backward fashion starting from a final marking which correspond to the made observation and ending up with initial source causes. Such a backward analysis is realized either by constructing a reachability graph or by invariant computation. The invariant technique presents the advantage of efficiency in terms of the storage and time complexity compared to the reachability graphs technique.

For experimental results, we have implemented the distributed analysis techniques (i.e. both the BW-Analysis and the P-invariant approach) on a local Ethernet network where each agent resides in a particular host of the network. The objective is to perform different series of experiment addressing the following issue: comparison between the distributed invariant approach to diagnosis and that based on reachability graphs (i.e. the distributed BW analysis approach) in terms of their running time. Actually, the experiment has been done on a fewer academic examples of small size (no more than 20 places and 20 transitions for each BPN). We considered several cases of malfunctions for each system model in such a way to consider all the main fault evolutions described in the model. Preliminary results of such a comparison showed a quite good behavior of the invariant based approach with respect to the reachability graphs approach.

Many issues remain to be investigated. Among those we mention:

- the possibility of using common transitions, instead of bordered places, between BPNs to model interactions among subsystems and then exploiting T-invariants rather than P-invariants as a structural technique.

- the adoption of a High-Level Net formalism such as Colored Petri Nets to represent causal models; in this case it could be possible to choose between performing analysis directly on it or to unfold the net into an ordinary Petri Net (this is possible because

each state or manifestation can assume a finite number of different instantiations) on which the proposed analysis techniques can be performed.

Bibliography

[1] A. Aghasaryan, E. Fabre, C. Jard, and A. Benveniste, A Petri net approach to fault detection and diagnosis in distributed systems. In *IEEE Conference on Decision and Control*, pp.702–731, San Diego, CA, 1997.

[2] C. Anglano and L. Portinale, B-W Analysis: A Backward Reachability Analysis for Diagnostic Problem Solving Suitable to Parallel Implementation, *Application and Theory of Petri Nets, LNCS*, vol.815, pp.39-58, 1994.

[3] P. Baroni, G. Lamperti, P. Pogliano, and M. Zanella, Diagnosis of large active systems. *Artif. Intell.*, 110(1):135–183, 1999.

[4] H. Bennoui, M. Maouche and M. Bettaz, Une approche pour le diagnostic des pannes dans les réseaux, *Proc. of the Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'99)*, Hermès Edition, Nancy, France, pp.433-445, 1999.

[5] H. Bennoui and A. Chaoui, Backward Reachability Analysis based on BPNs to Model-based Diagnosis with Relationships among Symptoms, *International Review on Computers and Software*, vol.4, no.2, pp.350-356, 2007.

[6] H. Bennoui, A. Chaoui and K. Barkaoui, Distributed Causal Model-based Diagnosis based on Interacting Behavioral Petri Nets, *Proc. of the* 8^{th} *International Symposium on Parallel and Distributed Computing*, Lisbon, pp.99-106, 2009.

[7] H. Bennoui, A. Chaoui and K. Barkaoui, Exploiting P-Invariant Analysis for Distributed Systems Diagnosis based on Interacting Behavioral Petri Nets, *Proc. of the ISIICT'09*, Amman, (eWiC series, British Computer Society), pp. 184-195, 2009.

[8] A. Benveniste, E. Fabre, C. Jard, and S. Haar. Diagnosis of asynchronous discrete event systems, a net unfolding approach. *IEEE TAC*, 48:714–727, 2003.

[9] C. D. Bocaniala and J. Sa da Costa, Novel framework for using causal models in distributed fault diagnosis, *Proc. of the Workshop on Advances in Control and Diagnosis*, Karlsruhe, Germany, pp.142-147, 2004.

[10] R. K. Boel and J. H. van Schuppen, Decentralized failure diagnosis for discrete-event systems with costly communication between diagnosers. In *International Workshop on Discrete Event Systems*, pp.175–181, 2002.

[11] M. P. Cabasino, Diagnosis and Identification of Discrete Event Systems using Petri Nets. PhD thesis, University of Cagliari, Italy, march 2009.

[12] J. Cardoso, L. A. Kunzle, and R. Valette, Petri net based reasoning for the diagnosis of dynamic discrete event systems. In *6th International Fuzzy Systemes Association World Congress*, pp.333-336, July 1995.

[13] C. G. Cassandras and S. Lafortune, Introduction to Discrete Event Systems, Second Edition, Springer, 2008.

[14] L. Console, T. Dupré, and P. Torasso, Abductive Reasoning through Direct Deduction from Completed Domain Models, In *Methodologies for Intelligent Systems 4*, Z. Raz editor, pp.175-182, North Holland, 1989.

[15] L. Console, A. J. Rivolin, T. Dupré, and P. Torasso, Integration of Causal & Temporal Reasoning in Diagnostic Problem Solving, *Avignon 89, Session 6*, pp.309-323, 1989.

[16] L. Console and P. Torasso, Hypothetical reasoning in causal models, *International Journal of Intelligent Systems*, vol.5, pp.83-124, 1990.

[17] L. Console and P. Torasso, Integrating Models of the Correct Behavior into Abductive Diagnosis, In *Proc. 9th ECAI*, Stockholm, pp.160-166, 1990.

[18] L. Console and P. Torasso, A Spectrum of Logical Definitions of Model-Based Diagnosis, *Computational Intelligence*, vol.7, no.3, pp.133-141, 1991.

[19] L. Console and L. Portinale, Model-based Diagnosis of System Malfunction with Petri Nets, In *Robotics and Flexible Manufacturing Systems*, S.G. Tzafestas and J.C. Gentina editors, pp.417-426, Elsevier Science, 1992.

[20] L. Console, L. Portinale, T. Dupré, and P. Torasso, Combining Heuristic and Causal Reasoning in Diagnostic Problem Solving, In *Second Generation Expert Systems*, J.M. Davis, J.P. Krivine, and R. Simmons editors, Springer Verlag, 1993.

[21] P. T. Cox and T. Pietrzykowski, General diagnosis by abductive inference, In *Symposium on Logic Programming*, pp.183-189, 1987.

[22] M. O. Cordier and A. Grastien, Exploiting independence in a decentralised and incremental approach of diagnosis, In Manuela M. Veloso, editor, *IJCAI*, pp.292–297, Hyderabad, India, 2007.

[23] R. Davis, Diagnostic Reasoning Based on Structure and Behavior, *Artificial Intelligence 24(1-3)*, pp.347-410, 1984.

[24] R. Davis and W. Hamscher, Model-based Reasoning: Troubleshooting, In *Exploiting Artificial Intelligence*, H. E. Shorbe editor, pp.297-346, Morgan Kaufmann, 1988.

[25] R. Debouk, S. Lafortune, and D. Teneketzis, Coordinated decentralized protocols for failure diagnosis of discrete event systems. *Journal of Discrete Event Dynamical Systems: Theory and Application*, 10:33–86, 2000.

[26] R. Debouk, S. Lafortune, and D. Teneketzis, On the effect of communication delays in failure diagnosis of decentralized discrete event systems, *Discrete Event Dynamic Systems*, 13(3):263–289, 2003.

[27] J. de Kleer and B. C. Williams, Diagnosing multiple faults, *Artificial Intelligence 32(1)*, pp.97-130, 1987.

[28] J. de Kleer and B. C. Williams, Diagnosis with behavioral modes, In *Proc. of the 11th IJCAI*, Detroit, pp.1324-1330, 1989.

[29] J. de Kleer, A. K. Mackworth, and R. Reiter, Characterizing diagnoses and systems, *Artificial Intelligence 56(2-3)*, pp.197-222, 1992.

[30] J. de Kleer, Fundamentals of Model-based Diagnsis, *Safe-Process*, 2005.

[31] O. Dressler and P. Struss, The Consistency-based Approach to Automated Diagnosis of Devices, In U. Gnowho ed., *A Great Collection in Studies in Logic, Language and Information*, CLSI Publications, pp.1-46, 1996.

[32] B. El-Ayeb, P. Marquis, and M. Rusonowitch, Preferring diagnosis by abduction, *IEEE Transactions 23(3)*, 1993.

[33] Y. El Fattah and P. O'Roke, Learning Multiple Fault Diagnosis, *Technical Report 91-06*, Department of Information and Computer Science, University of California, Irvine, CA, 1991.

[34] E. Fabre, A. Benveniste, and C. Jard. Distributed diagnosis for large discrete event dynamic systems. In *15th IFAC World Congress*, Barcelona, July 2002.

[35] E. Fabre, A. Benveniste, S. Haar, and C. Jard, Distributed monitoring of concurrent and asynchronous systems, *Discrete Event Dynamic Systems*, 15(1):33–84, 2005.

[36] E. Fabre and A. Benveniste, Partial order techniques for distributed discrete event systems: Why you cannot avoid using them. *Discrete Event Dynamic Systems*, 17(3):355–403, 2007.

[37] E. Garcia, A. C. Salvador, F. Morant, E. Q. Cucarella, and R. B. Giménez. Modular fault diagnosis based on discrete event systems. *Discrete Event Dynamic Systems*, 15(3):237–256, 2005.

[38] S. Genç and S. Lafortune, Distributed Diagnosis of Place-Bordered Petri Nets, *IEEE TASE*, vol.4, no.2, pp.206-219, 2007.

[39] A. Giua and C. Seatzu, Fault detection for discrete event systems using petri nets with unobservable transitions, In *44th Int. Conf. on Decision and Control and European Control Conference*, pp. 6323–6328, Seville, Spain, December 2005.

[40] W. Hamscher, Modeling Digital Circuits for Troubleshooting, *Artificial Intelligence*, 1991.

[41] L. J. Holtzblatt, Diagnising multiple failures using knowledge of component states, *IEEE Proc. on AI applications*, pp.139-143, 1988.

[42] S. Jiang and R. Kumar, Failure diagnosis of discrete event systems with lineartime temporal logic fault specifications. In *IEEE TAC*, pp.128–133, 2001.

[43] G. Jiroveanu and R. K. Boel, Petri Net model-based distributed diagnosis for large interacting systems, *Proc. of the DX'05*, pp.25-30, 2005.

[44] C. N. Hadjicostis and G. C. Verghese, Monitoring discrete event systems using Petri net embeddings. In *Proceedings of the 20th International Conference on Application and Theory of Petri Nets*, pp.188–207, London, UK, 1999. Springer-Verlag.

[45] H. Hu, A. Gehin, and M. Bayart, A merge method for decentralized discrete-event fault diagnosis, *International Conference on Internet Monitoring and Protection*, pp.119–124, 2008.

[46] G. Jiroveanu and R. K. Boel. Distributed contextual diagnosis for very large systems, In *Proceedings of WODES2004*, pp.1-8, 2004.

[47] G. Jiroveanu and R. K. Boel, A distributed approach for fault detection and diagnosis based on time Petri nets. *Math. Comput. Simul.*, 70(5):287–313, 2006.

[48] Y. Koseki, Experience Learning in Model-based Diagnostic Systems, In *Proc. of the 11th IJCAI*, Detroit, pp.1356-1362, 1989.

[49] R. Kumar and S. Takai, Inference-based ambiguity management in decentralized decisionmaking: Decentralized control of discrete event systems. *IEEE TAC*, 52(10):1783 –1794, 2007.

[50] R. Kumar and S. Takai. Inference-based ambiguity management in decentralized decisionmaking: Decentralized diagnosis of discrete-event systems. *IEEE TASE*, 6(3):479–491, 2009.

[51] S. Lafortune, D. Teneketzis, M. Sampath, R. Sengupta and K. Sinnamohideen, Failure Diagnosis of Dynamic Systems: An Approach based on Discrete Event Systems, *Proc. of the American Control Conf.*, pp.2058-2071, 2001.

[52] G. Lamperti and M. Zanella. Diagnosis of active systems – principles and techniques. 741, 2003.

[53] K. Lautenbach, S. Philippi and A. Pinl, Bayesian Networks and Petri Nets, *In E. Schnieder (Hrsg), Entwurf komplexer Autimatisierungssysteme*, EKA 2006, 9. Fachtagung, Braunschweig, 2006.

[54] X. Le Guillou, M. O. Cordier, S. Robin, and L. Rozé. Chronicles for on-line diagnosis of distributed systems. Research report, INARIA, 2008.

[55] J. McCarty, Applications of Circumscription to Formalizing Common-sense Knowledge, *Artificial Intelligence 28*, pp.89-116, 1986.

[56] M. A. Mejia, E. L. Mellado, A. R. Trevino, and I. R. Rangel, Petri net based fault diagnosis of DES. In *Proc. IEEE-. SMC*, pp.4730–4735, Washington, USA, 2003.

[57] R. Milne (ed.), Special issue on causal and diagnostic reasoning, *IEEE Transactions on Systems, Man and Cybernetics* 17(3), 1987.

[58] I. Mozetic, Hierarchical Model-based Diagnosis, *International Journal of Man-Machine Studies* 35, pp.329-362, 1988.

[59] T. Murata and D. Zhang, A Predicate-Transition Net Model for Parallel Interpretation of Logic Programs, *IEEE Transactions on Software Engineering*, SE 14(4):481-497, 1988.

[60] T. Murata, Petri Nets: Properties, Analysis and Applications, *Proc. of the IEEE* vol.77, no.4, pp.541-580, 1989.

[61] R. Patil, Causal representation of patient illness for electrolyte and acid-base diagnosis, *Technical Report MIT/LCR/TR-267*, MIT, Cambridge, MA, 1981.

[62] Y. Pencolé, Decentralized diagnoser approach: application to telecommunication networks, In *International Workshop on Principles of Diagnosis*, Michoacen, Mexico, 2000.

[63] Y. Pencolé, M.O. Cordier, and L. Roze, Incremental decentralized diagnosis approach for the supervision of a telecommunication network, In *IEEE Conference on Decision and Control*, Las Vegas, Nevada, USA, 2002.

[64] Y. Pencolé and M.O. Cordier, A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks, *Artificial Intelligence*, 164(1-2):121–170, May 2005.

[65] G. Peterka and T. Murata, Proof Procedure and Answer Extraction in Petri Net Model of Logic Programs, *IEEE TSE*, vol.15, no.2, pp.209-217, 1989.

[66] D. Poole, Normality and faults in logic-based diagnosis, In *Proc. of the 11th IJCAI*, Detroit, pp.1304-1310, 1989.

[67] L. Portinale, Exploiting T-invariant Analysis in Diagnostic Reasoning on a Petri Net Model, *Application and Theory of Petri Nets, LNCS*, vol.691, no.1, pp.339-356, 1993.

[68] L. Portinale, Improving Model-Based Diagnosis through Algebraic Analysis: the Petri Net Challenge, *Proc. of the* 13^{th} *National Conference on Artificial Intelligence (AAAI'96)*, pp.952-958, 1996.

[69] L. Portinale, Behavioral Petri Nets: A Model for Diagnostic Knowledge Representation and Reasoning, *IEEE TSMC–Part B: Cybernetics*, vol.27, no.2, pp.184-195, 1997.

[70] W. Qiu and R. Kumar, Decentralized failure diagnosis of discrete event systems. *IEEE TSMC, Part A*, 36(2):384–395, 2005.

[71] W. Qiu and R. Kumar. Distributed failure diagnosis under bounded delay using immediate observation passing protocol. In *American Control Conference*, Poland, 2005.

[72] W. Qiu and R. Kumar, A new protocol for distributed diagnosis. In *American Control Conference*, June 2006.

[73] J. A. Reggia, D. S. Nau, and P. Y. Wang, Diagnostic expert systems based on a set covering model, *International Journal of Man-Machine Studies* 19(5), pp.437-460, 1983.

[74] R. Reiter, A theory of diagnosis from first principles, *Artificial Intelligence 32(1)*, pp.57-96, 1987.

[75] S. L. Ricker and K. Rudie, Decentralized failure diagnosis with asynchronous communication between supervisors, In *Proc. of the IEEE Conference on Decision and Control*, 2001.

[76] N. Roos, A. T. Teije, A. Bos and C. Witteveen, Multi-Agent Diagnosis with spatially distributed knowledge, *Proc. of the BNAIC'02*, pp.275-282, 2002.

[77] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen and D. Teneketzis, Diagnosablity of Discrete Event Systems, *IEEE TAC*, vol.40, no.9, pp.1555-1575, 1995.

[78] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D.C. Teneketzis. Failure diagnosis using discrete-event models. *IEEE Transactions on Control Systems Technology*, 4(2):105–124, Mar 1996.

[79] M. Schroeder and G. Wagner, Distributed diagnosis by vivid agents, In *Proc. of the first conference on Autonomous Agents*, pp.268–275, 1997.

[80] E. J. Shortliffe, Computer Based Medical Consultations: MYCINE, *Elsevier Science*, New York, 1976.

[81] V. S. Srinivasan and M. A. Jafari, Fault detection/monitoring using timed Petri nets. *IEEE TSMC*, 23(4):1155-1162, 1994.

[82] R. Su and W. M. Wonham, A model of component consistency in distributed diagnosis, In *Proc. IFAC Workshop on Discrete Event Systems (WODES'04)*, pp.427–432, Reims, France, 2004.

[83] R. Su, W. M. Wonham, J. Kurien, and X. Koutsoukos, Distributed diagnosis for qualitative systems, In *WODES'02*, 2002.

[84] R. Su and W. M. Wonham, Hierarchical fault diagnosis for discrete-event systems under global consistency, *Discrete Event Dynamic Systems*, 16(1):39–70, 2006.

[85] S. Tripakis, Fault diagnosis for timed automata, In Werner Damm and Ernst-Rüdiger Olderog, editors, FTRTFT, volume 2469 of *Lecture Notes in Computer Science*, pp.205–224, 2002.

[86] T. Ushio, L. Onishi, and K. Okuda, Fault detection based on petri net models with faulty behaviors. In *Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics*, pp.113–118, San Diego, USA, 1998.

[87] Y.Wang, T.S. Yoo, and S. Lafortune. New results on decentralized diagnosis of discrete-event systems, In *Annual Allerton Conference*, 2004.

[88] Y. Wang, T.S. Yoo, and S. Lafortune, Diagnosis of discrete event systems using decentralized architectures. *Discrete Event Dynamic Systems*, 17(2):233–263, 2007.

[89] S. Weiss, C. Kulikowski, S. Amarel, and A. Safir, A model based method for computer-aided medical decision making, *Artificial Intelligence* 11(1-2), pp.145-172, 1978.

Abstract

This thesis deals with the problem of distributed causal model-based diagnosis on interacting Behavioral Petri Nets (BPNs). The system to be diagnosed comprises different interacting subsystems (each modeled as a BPN) and the diagnostic system is defined as a multi-agent system where each agent is designed to diagnose a particular subsystem on the basis of its local model, the local received observation and the information exchanged with the neighboring agents. The interactions between subsystems are captured by tokens that may pass from one net model to another via bordered places. The diagnostic reasoning scheme is accomplished locally within each agent by exploiting classical analysis techniques of Petri nets like reachability graph and invariant analysis. Once local diagnoses are obtained, agents begin to communicate to ensure that such diagnoses are consistent and recover completely the results that would be obtained by a centralized agent having a global view about the whole system.

Keywords: model-based diagnosis, causal models, Petri nets, reachability analysis, invariant analysis.

Résumé

Cette thèse traite le problème de diagnostic à base de modèles causaux par réseaux de Petri comportementaux (BPNs). Le système à diagnostiquer est considéré comme une collection de sous-systèmes en interaction (chacun est modélisé comme un BPN) et le système de diagnostic est défini comme un système multi-agents où chaque agent est chargé de diagnostiquer un sous-système particulier en se basant sur son modèle local, l'observation locale reçue et les informations échangées avec les agents voisins. Les interactions entre les sous-systèmes sont capturées par des jetons qui peuvent passer d'un modèle à l'autre via des places communes entre modèles BPNs. Le mécanisme de résolution est accompli localement au niveau de chaque agent par exploitation des techniques classiques d'analyse des réseaux de Petri comme l'analyse à base de graphes d'atteignabilité et celle basée sur les invariants. Une fois les diagnostics locaux sont obtenus, les agents entrent dans une étape de communication afin d'assurer que ces diagnostics sont cohérents et recouvrent les résultats obtenus par un agent centralisé ayant une vision globale autour du système entier.

Mots-clés : diagnostic basé-modèle, modèles causaux, réseaux de Petri, analyse d'atteignabilité, analyse d'invariants.